I.J. Intelligent Systems and Applications, **2017, 3, 26-33**
Published Online March 2017 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijisa.2017.03.04

Using Machine Learning Techniques to Support Group Formation in an Online Collaborative Learning Environment

Elizaphan M. Maina
Department of Computing and Information Technology, Kenyatta University, P.O. Box 43844, 00100, Nairobi, Kenya
E-mail: maina.elizaphan@ku.ac.ke

Robert O. Oboko and Peter W. Waiganjo
School of Computing and Informatics University of Nairobi, P. O. Box 30197, 00100, Nairobi, Kenya
E-mail: roboko@uonbi.ac.ke; waiganjo@uonbi.ac.ke

Abstract**—**The current Learning Management Systems used in e-learning lack intelligent mechanisms which can be used by an instructor to group learners during an online group task based on the learners' collaboration competence level. In this paper, we discuss a novel approach for grouping students in an online learning group task based on individual learners' collaboration competence level. We demonstrate how it can be applied in a Learning Management System such as Moodle using forum data. To create the collaboration competence levels, two machine learning algorithms for clustering namely Skmeans and Expectation Maximization (EM) were applied to cluster data and generate clusters based on learner's collaboration competence. We develop an intelligent grouping algorithm which utilizes these machine learning generated clusters to form heterogeneous groups. These groups are automatically made available to the instructor who can proceed to assign them to group tasks. This approach has the advantage of dynamically changing the group membership based on learners' collaboration competence level.

Index Terms**—**Learning Management Systems, Online Group Task, Collaboration Competence Level, Intelligent grouping Algorithm, Machine learning.

I. Introduction

Intelligent techniques like Machine Learning (ML) can be used to analyze online collaboration activities thus, providing data which can be applied by the instructor to improve the collaboration process [1]–[6]. Moreover, research on the clustering of collaboration data including [1], [6], [7] has revealed that ML techniques can be applied to analyze students interaction in group work and rank learners according to their collaboration level. This helps learners and tutors to evaluate collaborative work and identify possible problems as they arise. However, these studies do not address the aspect of group formation which can impact on group performance. Without appropriate support in group formation, students tend to form groups which are more social, ignoring aspects of collaboration competence. For example, self created groups tend to be more associated with demographic characteristics while randomly created groups could be homogenous rather than heterogeneous in terms of individual capabilities. Moreover, current research does not suggest an algorithm which can group students based on their collaboration competence level. These aspects give impetus to exploration of group formation methods based on collaboration competence data and to provide intelligent support in group formation for online collaborative learning.

In this paper, we demonstrate how clustering can be achieved using discussion forum data in a Learning Management Systems (LMS) such as Moodle, and later to form clusters based on learners' collaboration competence level. Through an intelligent grouping algorithm these clusters are applied in the automatic formation of heterogeneous groups. Currently, Moodle can only group students automatically through random methods. The random group member assignment method is popular in Moodle because it does not consume a lot of time, but the level of heterogeneity may not match the diversity in learning capabilities which is required in a group. Furthermore, Machine Learning tools which can analyze discussion forums are yet to be integrated in Moodle. To that end, this study implements an intelligent grouping algorithm which does not consume a lot of time (requires little intervention by the instructor) and is able to create heterogeneous groups with diverse collaboration competence levels.

The paper is organized as follows: Section II discusses work related to group formation, techniques for analyzing collaboration data in group work and clustering methods, Section III discusses methodology used in the design and implementation of the intelligent grouping algorithm. Section IV presents sample results on the intelligent grouping algorithm in realistic e-learning environment and finally, section V draws conclusions and presents future directions.

II. Related Work

A. Group Formation

Group formation is the process of identifying students and assigning them to a specific group so that they belong to one specific group when doing a group task [8]. The criteria for selecting members in a group do affect the members' commitment. Group members who choose fellow group members are more relationally satisfied with their group and more committed to work together than members who are randomly assigned to a group [9]. The random selection method is highly utilized by instructors due to the ease of implementation and 'fair' distribution, which gives a student equal chance to be a member of any group, hence both social and academic heterogeneity can somehow be achieved. However, it can also lead to lack of diversity in skills within the group [10]. Randomly selected group members have also been proven to utilize their time during group meeting more effectively and are more task oriented probably because, familiarity among members is less which makes the groups' social network less compared to self-selected groups [11]. Despite these advantages, random selection has proved to be less effective in improving group performance, it leads to inferior group dynamic ratings, and results in a higher degree of conflicts [11]. Hence, there is need to explore other techniques such as those doing intelligent grouping, which are more dynamic and are capable of considering collaboration competences among learners.

The use of intelligent systems to do group formation in online collaborative learning environments has also been reported in recent research [5], [6], [12]. Although computer based random selection methods have been preferred for large classes of learners, intelligent techniques are better because they do incorporate learner's characteristics such as learning style [12], learner's profile and context [13] and contextual information [5] and they can dynamically change member allocation to groups. The ability to change the group member composition in real time enables the leveling up of learning results and improvements in the participants' social relationships. Some of the intelligent techniques have applied the use of Machine Learning techniques such as Instance-based Learning and Bayesian network which are capable of using contextual information to learn the user behavior and predict an appropriate group for the learner based on the contextual information [5], [12]. However, the application of learners' collaboration competences in group formation is yet to be explored adequately in online collaborative learning environments.

B. Analyzing Collaboration

Collaboration can be characterized by three important elements: independence [14], interdependence [15] and synthesis of information [16] . The three elements work together for effective online collaborative learning. Independence can be analyzed by measuring the extent of influence of the instructor or other participants on individual participation and interaction. Individuals who post new ideas rather than just replies are more independent hence, more collaborative. Interdependence, on the other hand, requires active participation by each member. Participation can be measured by counting the number of messages and statements submitted by each individual and the group as responses to the other participants' posts. This allows both groups and individuals to be compared in terms of their level of participation. Synthesis can be measured in two ways. Firstly, by the interaction pattern of the discussion that occurs when a participant contributes a statement, and another participant synthesizes it by extending the idea and subsequent messages yields new ideas. This requires content analysis of the individual thread contributed in the discussion forum. Secondly, synthesis can be analyzed by examining the relationship between original comments and the final product. In this study, we apply the latter approach where the instructor compares an individual post with the final product and assigns a numerical value according to the relevance of the post to the final product, which we refer to as forum rating. This in turn can tell us the level of individual contribution in relation to the final product.

By analyzing discussion forum data in terms independence, interdependence and synthesis it is possible to assign learners into different collaboration competence levels. In the light of these arguments, in this study three attributes namely forum posts, forum replies and forum rating have been applied to define three collaboration competence levels. These are High, Medium and Low, characterized by different levels of interdependence, synthesis and independence [17]. Table 1 illustrates the characteristics associated with the three levels of collaboration.

Table 1. Characteristics Associated with Collaboration Competence Levels (CCL)

CCL	Characteristics
High	If a student logs-in often and participates and interacts actively, his/her profile is clearly collaborative and the learner can be ranked into a higher level of collaboration competence.
Medium	If a student logs-in often and participates and interacts moderately, his/her profile is medium and the learner can be ranked into a medium level of collaboration competence.
Low	If a student logs-in and participates rarely and there is no indication of interdependence, synthesis and independence, his profile is non-collaborative and the learner can be ranked into a low level of collaboration competence.

C. Clustering Algorithms

Clustering is the process of finding out a group of objects which have similar characteristics and assigning them to a cluster/group such that objects in the same cluster are similar in some sense. Clustering is a method of unsupervised Machine Learning, and a common technique for statistical data analysis. The principle of clustering is maximizing the similarity within the object groups in the cluster and minimizing the differences between the object groups in that cluster [18]. Clustering methods can be classified into different types [19], including hierarchical (single-link, complete-link, etc.) and objective-function-based algorithms (*Skmeans*, *Expectation Maximization (EM)*, etc.). These clustering algorithms are available in Weka software which is open source software implemented in Java code and is platform independent. In the Weka workbench, the algorithms can be applied directly to a dataset or invoked through other software [20]. The Weka workbench also provides a graphical interface which allows easy visualization of data and also provides other explorers for managing data.

In e-learning, clustering can be used to group students according to their collaboration competence level in a collaborative learning environment [7], predict their academic performance [21] and group students in order to give them differentiated guidance according to their learning skills and other characteristics [22]. In this study, we demonstrate how *Skmeans* and Expect*ation Maximization (EM)* clustering algorithms can be used to cluster students based on their collaboration competence level. Through intelligent grouping, the resulting clusters are utilized to form heterogeneous groups with diverse collaboration competence levels.

III. Methodology

In this section, we introduce a multi-methodological approach that was used in developing this intelligent grouping algorithm. This system development methodology consists of four research strategies: theory building, experimentation, observation and system development as illustrated in Fig. 1. In this methodology, system development is viewed as the hub of research that interacts with other research methodologies to form an integrated and dynamic research process [23]. In case of complex research areas such as intelligent systems, multi-methodological approach becomes an effective strategy for gaining a complete understanding of the system [23]. For the purpose of this paper we only discuss the system development stage in terms of:

A. System architecture
B. Cluster implementation
C. Intelligent grouping algorithm

Moodle has been utilized in this study as a LMS since it is open source software, which makes it possible to customize the source code and it is also widely used in institutions of higher learning.

A. System Architecture

In this section, we demonstrate a system architecture that integrates Machine Learning (ML) algorithms into LMSs such as Moodle. . In this architecture the learner is required to interact with the LMS through discussion forums. Data generated during a discussion forum is stored in the Moodle database. In order to use ML to support discussion forums in Moodle, first the system architecture for Moodle is linked to a ML environment. The ML environment contains the clustering algorithms which are applied to the preprocessed Moodle forum data obtained from Moodle Database (DB) to create clusters which are equivalent to the number of collaboration competence levels defined by the instructor. The data for the resulting clusters is post-processed and stored back to Moodle DB. This cluster data is applied by the intelligent grouping algorithm to create groups for collaborative work. Fig. 2 illustrates this system architecture

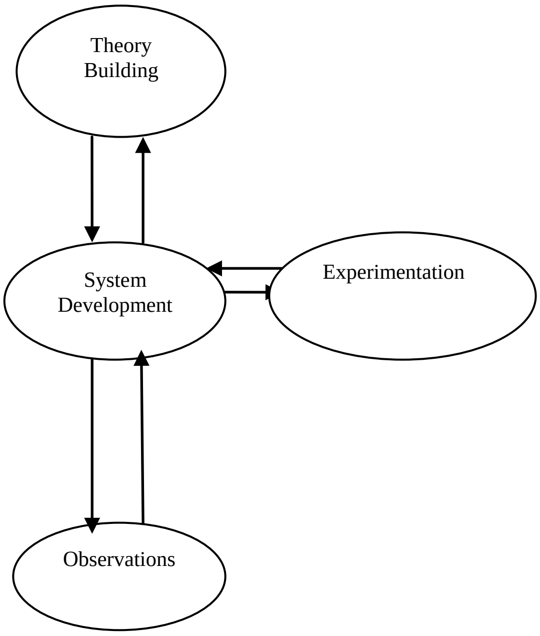


Fig.1. Multi-methodological approach to system development

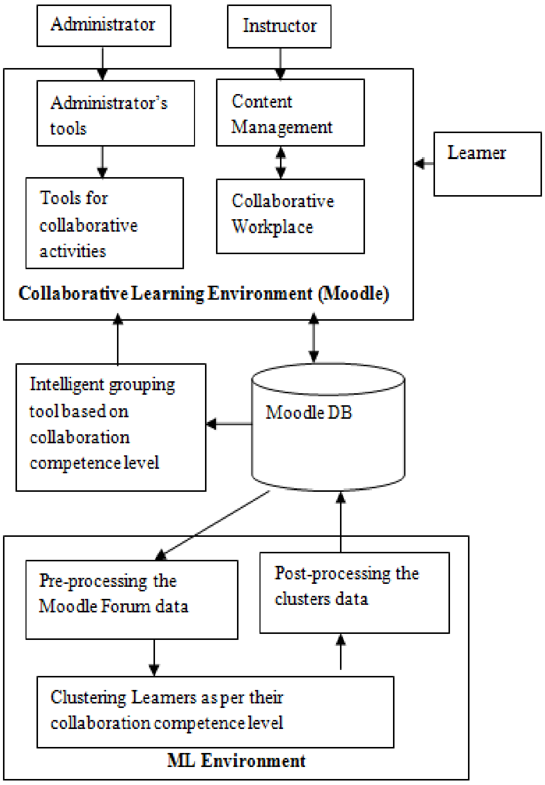

Administrator
Instructor
Administrator's tools
Content Management
Learner
Tools for collaborative activities
Collaborative Workplace
Collaborative Learning Environment (Moodle)
Intelligent grouping tool based on collaboration competence level
Moodle DB
Pre-processing the Moodle Forum data
Post-processing the clusters data
Clustering Learners as per their collaboration competence level
ML Environment

Fig.2. System Architecture for ML support to group work

B. *Cluster Implementation*

The Weka workbench has several clustering algorithms available. However, in this study *Skmeans* and *EM* clustering algorithms have been applied. *Skmeans* has advantage of being computationally faster when dealing with large number of variables than hierarchical clustering provided the value of k is small [24]. On the other hand, *EM* has the advantage of being able to estimate data distribution when data is partially missing or hidden [25]. In this study, the objective was to group students into 3 clusters based on discussion forum data in Moodle. These three clusters represented the three categories of collaboration competence levels namely Higher, Medium and Low. Therefore, this section discusses how the forum data is pre-processed and fed into Weka.PHP program.

The forum data in Moodle is stored in MySQL Moodle database. Although forum data have many attributes, we have utilized three attributes which possess data that corresponds to the three indicators of collaboration. The first attribute is a new post, which is an original idea; the second is a reply to a post, which corresponds to a response to an existing idea and the third is average rating of the posts, which is done by an instructor and it indicates the level of relevance of the post on the issues under discussion.

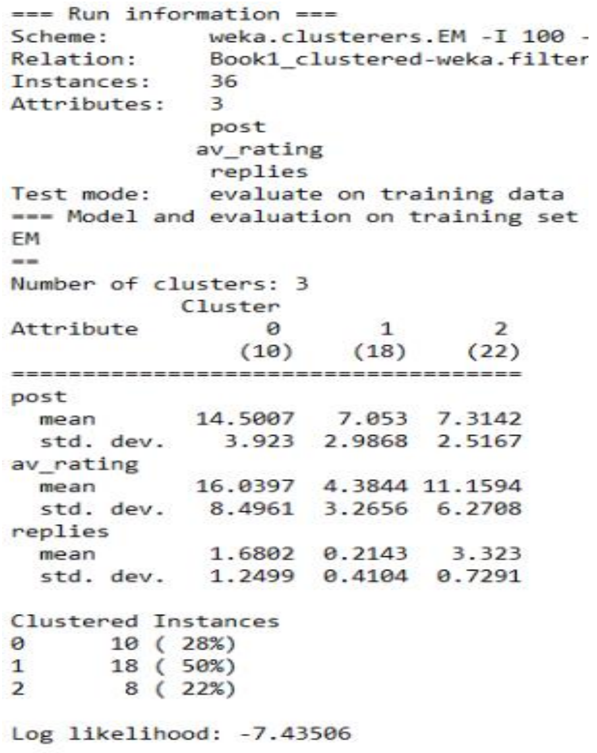
Preprocessing the data requires the data to be cleaned and transformed into an appropriate form which can be processed by Weka clustering algorithms. Moodle forum data and forum rating is stored in the following tables: mdl_forum, which stores information about all forums; mdl_forum_posts, which stores all posts to the forums; mdl_forum_discussions, which stores all forums' discussions and mdl_rating, which stores the average rating of the posts. Since the data is stored in a Relation Database Management System (RDBMS), less cleaning and pre-processing is required and for our case, we only create a summarization table with the required fields from the above tables and export the result to a text file. The summary table is stored as text file with .cvs extension and it has the following columns: (i) User id, (ii) Number of posts, (iii) Number of replies and (iv) Forum ratings. This summary table is fed as an input to the Weka.PHP program which has the clustering algorithms. The Weka.jar library is invoked within the Weka.PHP page in Moodle. The Weka program takes the following input parameters: Input file, Type of Clustering (Skmeans or EM) and number of clusters. For this study, three clusters were formed to represent three different collaborative competence levels namely High, Medium and Low.

In order to establish whether cluster execution in Moodle was working, students in three different classes were given a discussion forum to discuss for a period of two weeks. After two weeks, the forum statistics which included *number_of_posts, number_of_replies* and *forum_ratings* were transformed into an attribute-relation file format *('testdata.arff')*. The dataset file was run in Weka software using these two clustering algorithms (Skmeans or EM). In Weka software, clustering using SKmeans and EM requires first a number of tests to be run, and then establish the values of two important parameters (seed value and maximum alteration). In this study, cross validation was done with the dataset file for both SKmeans and EM in order to establish the best values for these parameters which could give results with high accuracy level. For Skmeans, the best seed value was 10 and the maximum alteration value was 500, while for EM best seed value was 500 and the maximum alteration value was 100. Using these parameters, the data set ('testdata1.arff') file which contained three data sets for the forum summary data for the three classes was run in Weka. Table 2 shows the summary results; where N is the number of students, C is the cluster type and Sk is the Skmeans algorithm.

Table 2. Summary Results in Weka for Skmeans and EM in 3 Different Datasets

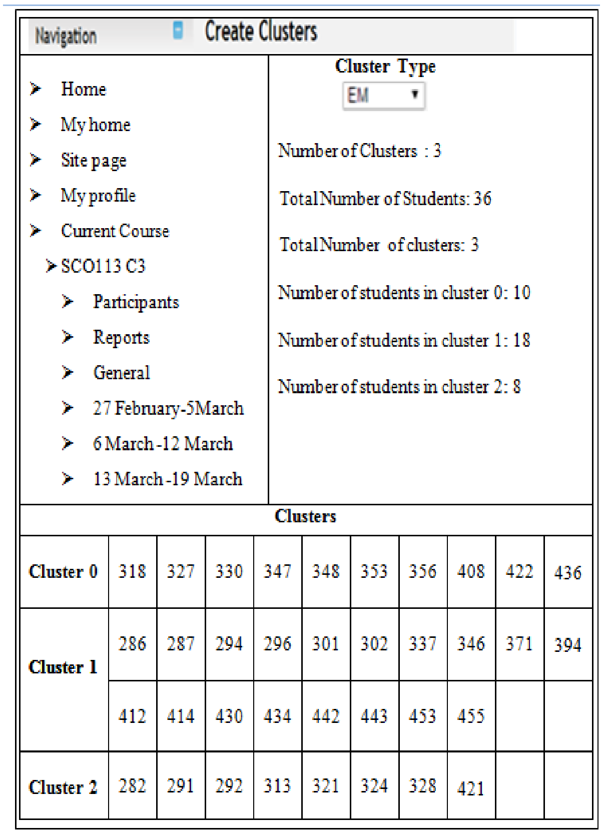
N	36		109		151	
C	Sk	EM	Sk	EM	Sk	EM
0	9 (25%)	10 (28%)	12 (11%)	61 (56%)	35 (23%)	44 (29%)
1	6 (17%)	18 (50%)	39 (36%)	14 (13%)	35 (23%)	77 (51%)
2	21 (58%)	8 (22%)	58 (53%)	34 (31%)	81 (54%)	30 (20%)

In the Table 2, Cluster 0 is High, Cluster 1 is Medium and Cluster 2 is Low. From Table 2, we observe that both SKmeans and EM almost gave similar distribution patterns on the number of students in different clusters regardless of the total number of students involved. We find that in every set of data, there is a cluster with a high number of students and one with a lower number of students regardless of the type of clustering algorithm applied. However, this distribution pattern does not correspond with cluster values for both algorithms. Through expert analysis, we found that a cluster with low values had students who had a high number of posts, replies and average ratings. Therefore, ranking was required to be done before using the cluster results to determine the best students who can be assigned as group mentors in their groups. To confirm that the cluster module was working perfectly, the three data sets which were executed in Weka software were used to do clustering in Moodle and the two results were compared. The same parameters were also applied in both cases. After a number of tests, it was found that the results from Moodle concurred with those obtained in Weka software both in terms of number students in each cluster and also in terms of cluster assignment for each student. Fig. 3 below shows results for EM clustering algorithm in Weka software for three clusters. This result concurs with the once shown in Fig. 4 in Moodle where cluster (0) had 10 students, cluster (1) had 18 students and cluster (2) had 8 students.



```
=== Run information ===
Scheme:        weka.clusterers.EM -I 100 -
Relation:      Book1_clustered-weka.filter
Instances:     36
Attributes:    3
               post
              av_rating
               replies
Test mode:     evaluate on training data
=== Model and evaluation on training set
EM
==
Number of clusters: 3
            Cluster
Attribute        0         1         2
              (10)      (18)      (22)
====================================
post
  mean       14.5007   7.053  7.3142
  std. dev.    3.923  2.9868  2.5167
av_rating
  mean       16.0397  4.3844 11.1594
  std. dev.   8.4961  3.2656  6.2708
replies
  mean        1.6802  0.2143   3.323
  std. dev.   1.2499  0.4104  0.7291

Clustered Instances
0       10 ( 28%)
1       18 ( 50%)
2        8 ( 22%)

Log likelihood: -7.43506
```

Fig.3. Screen shot showing testing results for EM clustering Algorithm sin Weka Software



Cluster Type

EM

Number of Clusters : 3

Total Number of Students: 36

Total Number of clusters: 3

Number of students in cluster 0: 10

Number of students in cluster 1: 18

Number of students in cluster 2: 8

Clusters

Cluster 0	318	327	330	347	348	353	356	408	422	436
Cluster 1	286	287	294	296	301	302	337	346	371	394
	412	414	430	434	442	443	453	455		
Cluster 2	282	291	292	313	321	324	328	421		

Fig.4. Screen shot showing testing results for EM clustering Algorithm in Moodle

In order to execute the clustering module in Moodle, a custom *'Discussion'* block is created to view and manage the clustering algorithms from the Weka.PHP program. The custom block has the cluster option which can be accessed by the instructor in his/her course. The cluster option is supposed to load the Weka.PHP program which provides the user an interface for creating the clusters and a display form which loads the cluster instances with students' identities e.g. 282, 327 and 408 in Fig. 4. Fig. 4 shows clustering results in Moodle for the first data set which had 36 students using EM algorithm.

C. *Intelligent Grouping Algorithm*

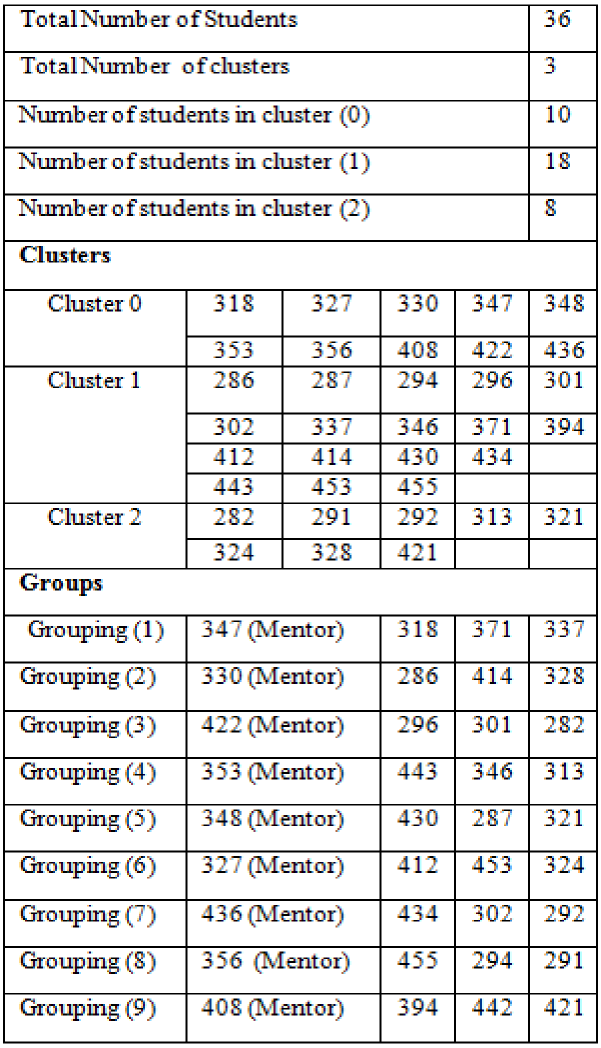
Data stored in these clusters was used to form heterogeneous groups using an intelligent grouping algorithm. To create heterogeneous groups, the data stored in the three collaborative competence levels (Cluster 0, Cluster 1 and Cluster 2) is converted to an array with *'userid'* values. A randomizing algorithm created using php *'randomarray'* function takes the array as input and produces an output array with randomized *'userid'* values. For example, if Cluster 0 corresponds to higher collaborative level and has *'userid'* values as per this order: 12, 34, 56, 23, 47 then after randomization the order changes to: 34, 47, 23, 56, 12. This randomization task is done for all clusters and then *'userids'* are ranked from Cluster 0 (most collaborative) to Cluster 2 (least collaborative). The result is stored in an array called *'rankedArray'*. It's from the *'rankedArray'* that the algorithm picks students from different collaborative levels as per the rank and assigns them to one group as per the specified group size. This process is performed iteratively until all students are assigned to a group. Students who are most collaborative are assigned a mentor role in their group.

The following pseudocode was applied to implement the intelligent grouping algorithm based on clustered data.

```
start_session := load_csv_file <-
filename(mdl_cluster_temp)
declare variable and initialize()<- inputs
int(i,j,n,a,b,no_of_cluster,userst,no_of_groups,rank)
declare variable and initialize()<- inputs
array(random_array,  new_array,test_array,group_array)

// store cluster assignment in double dimensional

array:(Array[i][j])
foreach(no_of_cluster);
userst=<get_recordset>;
  foreach(userst)
    Array[i][j]=userst;
      j++; i++;
for(i=0;i<n;i++)
random_array=Array[i];
shuffle(random_array);

// randomize the array by using shuffle function

test_array[]=random_array;

// assign members to groups
for(a=0;a<no_of_groups;a++)
for(b=0;b<sizeof(test_array[a]);b++)
        new_array[]=test[a][b];
for(c=0;c<no_of_groups; c++)        for(rank=c;
    rank<sizeof(new_array);rank+=no_of_groups)
        group_array[c][rank]=new_array[rank];
add_group_data_into_Moodle:= mdl_groups_members
        <- input(group_array[c][rank])
            exit_session()
```

IV. Testing Results and Discussion

In order to test the intelligent grouping algorithm, a group task in the form of a discussion forum in Moodle was given to a class of 36 students. The students were randomly placed in groups of four and they were required to discuss the group task online for a period of two weeks. After the two weeks, the forum statistics for each student in the form of number of posts, number of replies and average ratings were generated and stored in text file. The text file was fed to the clustering algorithms as described in section 3.3 to generate three clusters based on learner's collaboration competence level. The clusters from the *SKMeans* and *EM* clustering algorithms were found to be very similar. The resulting three clusters were applied in the intelligent grouping algorithm to generate nine heterogeneous groups with diverse collaboration competence levels where the number of students per group was defined as four. These groups were then automatically availed in the grouping module in Moodle and the instructor could assign them to a discussion forum or any other group activity as desired. Fig. 5 shows a screen short for the testing results in Moodle. To conceal student identity, automatically generated numbers were used to represent student's identity.



Total Number of Students					36
Total Number of clusters					3
Number of students in cluster (0)					10
Number of students in cluster (1)					18
Number of students in cluster (2)					8
Clusters					
Cluster 0	318	327	330	347	348
	353	356	408	422	436
Cluster 1	286	287	294	296	301
	302	337	346	371	394
	412	414	430	434	
	443	453	455		
Cluster 2	282	291	292	313	321
	324	328	421		
Groups					
Grouping (1)	347 (Mentor)		318	371	337
Grouping (2)	330 (Mentor)		286	414	328
Grouping (3)	422 (Mentor)		296	301	282
Grouping (4)	353 (Mentor)		443	346	313
Grouping (5)	348 (Mentor)		430	287	321
Grouping (6)	327 (Mentor)		412	453	324
Grouping (7)	436 (Mentor)		434	302	292
Grouping (8)	356 (Mentor)		455	294	291
Grouping (9)	408 (Mentor)		394	442	421

Fig.5. Sample results from intelligent grouping algorithm in Moodle

From Fig. 5, it can be observed that the algorithm distributed the students in such a way that each group is assigned four students who are members of different clusters hence, creating heterogeneous groups based on learners' collaboration competence level. In addition, Students who are in Cluster 0 (highly collaborative cluster) are assigned a mentor role in their group membership as this cluster constitutes highly collaborative members. The term *'mentor'* indicated that they would play the mentor role during discussions. Therefore, the testing confirmed that the intelligent grouping algorithm was capable of forming heterogeneous groups based on ranked clustered data.

V. Conclusion and Future Work

This paper has discussed a novel approach for grouping students based on collaboration competence level, with groups being created through clustering techniques. The implementation of the intelligent grouping algorithm in LMS such as Moodle suggests that the existing group formation techniques can be improved through Machine Learning techniques. The utilization of Machine Learning techniques to support group formation is timely since most of the institutions of higher learning in Kenya are faced with the challenge of providing adequate instructor support in blended e-learning[26]. This intelligent grouping algorithm in Moodle requires little intervention by the instructors when providing instructional support on the utilization of forums. This becomes a major advantage to those instructors who have little time to provide instructional support in online collaborative learning. This grouping mechanism can also be extended to other LMS such as Blackboard.

While this study only focused on two clustering algorithms and three attributes, there is a need to evaluate the performance of other clustering algorithms and the use of additional attributes about each learner in a discussion forum, and perhaps even give an indication on the optimal number of attributes for such studies. Additionally, further research needs to be carried out to evaluate the effectiveness of this grouping algorithm through an experimental design in a real e-learning environment.

ACKNOWLEDGMENT

The authors would like to acknowledge the National Commission for Science, Technology and Innovation, Kenya for funding the research.

REFERENCES

[1] A. R. Anaya and J. G. Boticario, “Clustering Learners according to their Collaboration.,” *Proc. 13th Int. Conf. Comput. Support. Coop. Work Des.*, 2009.

[2] A. R. Anaya and J. G. Boticario, “Ranking Learner Collaboration according to their Interactions.,” *1st Annu. Eng. Educ. Conf.*, 2010.

[3] A. R. Anaya and J. G. Boticario, “Content-free Collaborative Learning Modeling Using Data Mining. Special issue of User Modeling and User-Adapted Interaction on Data Mining in Education.,” 2011.

[4] B. M. McLaren, O. Scheuer, and J. Miksatko, “Supporting collaborative learning and e-Discussions using artificial intelligence techniques,” *Int. J. Artif. Intell. Educ.*, 2010.

[5] R. Messeguer, E. Medina, D. Royo, L. Navarro, and J. P. Juarez, “Group Prediction in Collaborative Learning,” in *In Intelligent Environments (IE), Sixth International Conference*, 2010, pp. 350–355.

[6] Y. Awuor and R. Oboko, “Automatic assessment of online discussions using text mining,” *Int. J. Mach. Learn. Appl.*, vol. 1, no. 1, p. 7–pages, 2012.

[7] B. N. L. Valetts and R. Gesa, “Modelling Collaborative Competence Level Using Machine Learning Techniques,” *In e-Learning*, pp. 56–60, 2008.

[8] M. Wessner and H. R. Pfister, “Group formation in computer-supported collaborative learning,” in *In Proceedings of the 2001 international ACM SIGGROUP conference on supporting group work, ACM*, 2001, pp. 24–31.

[9] J. Scott, *Social Network Analysis : a handbook*, 2nd ed. London: Sage, 2001.

[10] D. R. Bacon, K. A. Stewart, and E. S. Anderson, “Methods of assigning players to teams: A review and novel approach,” *Simul. Gaming*, vol. 32, no. 1, pp. 6–17, 2001.

[11] K. J. Chapman, M. Meuter, D. Toy, and L. Wright, “Can’t we pick our own groups? The influence of group selection method on group dynamics and outcomes,” *J. Manag. Educ.*, vol. 30, no. 4, pp. 557–569, 2006.

[12] S. Liu, M. Joy, and N. Griffiths, “iGLS: intelligent grouping for online collaborative learning,” in *In Advanced Learning Technologies, 2009. ICALT 2009, Ninth IEEE International Conference*, 2009, pp. 364–368.

[13] M. Muehlenbrock, “Learning group formation based on learner profile and context,” *Int. J. E-learning*, vol. 5, no. 1, pp. 19–24, 2006.

[14] J. Laffey, T. Tupper, D. Musser, and J. Wedman, “A computer-mediated support system for project-based learning,” *Educ. Technol. Res. Dev.*, 1998.

[15] D. W. Johnson, R. T. Johnson, and K. A. Smith, “Cooperative learning returns to college,” *Change*, vol. 30, no. 4, pp. 26–35, 1998.

[16] A. Kaye, “Learning together apart. In A. R. Kaye,” *Collab. Learn. Through Comput. Conf.*, 1992.

[17] E. Muuro, P. W. Wagacha, and R. Oboko, “Models for Improving and Optimizing Online and Blended Learning in Higher Education,” Jared Keengwe and J. J. Agamba, Eds. IGI Global. Pennsylvania, USA, 2014, pp. 204–219.

[18] C. Romero, S. Ventura, and E. Garcia, “Data mining in course management systems: Moodle case study and tutorial,” *Comput. Educ.*, vol. 51, no. 1, pp. 368–384, 2008.

[19] A. K. Jain, M. N. Murty, and P. J. Flynn, “Data Clustering: A Review,” *ACM Comput. Surv.*, vol. 31, no. 3, pp. 264–323, 1999.

[20] N. Sharma, A. Bajpai, and M. R. Litoriya, “Comparison the various clustering algorithms of Weka tools,” *Int. J. Emerg. Technol. Adv. Eng.*, vol. 2, no. 5, May 2012.

[21] R. Asif, A. Merceron, and M. K. Pathan, “Predicting student academic performance at degree level: a case study,” *Int. J. Intell. Syst. Appl.*, vol. 7, no. 1, p. 49, 2014.

[22] W. Hamalainen, J. Suhonen, E. Sutinen, and H. Toivonen, “Data mining in personalizing distance education courses,” in *In World Conference on Open Learning and Distance Education*, 2004, pp. 1–11.

[23] J. F. Nunamaker, J. R. M. Chen, and T. D. M. Purdin, “Systems Development in Information Systems Research,” *J. Manag. Inf. Syst.*, vol. 7, no. 3, pp. 89–106, 1991.

[24] R. T. Aldahdooh and W. Ashour, “DIMK-means‘ Distance-based Initialization Method for K-means Clustering Algorithm,’” *Int. J. Intell. Syst. Appl.*, vol. 5, no. 2, p. 41, 2013.

[25] S. P. Algur and P. Bhat, “Web Video Object Mining: A Novel Approach for Knowledge Discovery,” *Int. J. Intell. Syst. Appl.*, vol. 8, no. 4, 2016.

[26] E. Muuro, P. Wagacha, R.Oboko, and J. M. Kihoro, “Students’ Perceived Challenges in an Online Collaborative Learning Environment: A Case of Higher Learning Institutions in Nairobi, Kenya,” *Int. Rev. Res. Open Distance Learn.*, vol. 15, 2014.

Authors’ Profiles

Dr. Elizaphan M. Maina is a Lecturer in the Department of Computing and Information Technology at Kenyatta University, Kenya. He received B.ED Science degree from Kenyatta University, Kenya in 2000, Postgraduate Diploma in Computer Science from University of Nairobi, Kenya in 2002, Msc. in Information Systems from University of Nairobi, Kenya in 2005 and PhD in Computer Science from University of Nairobi, Kenya in 2015. He has taught computer Science for ten years in Universities and Tertiary colleges. Currently He has published journals in the field of artificial intelligence and e-learning. He lecturers in the field of artificial intelligence, programming, database systems and use of ICT in Education. His primary research focus is on integration of artificial intelligence techniques in e-learning in order to create new e-learning pedagogies which can support personalized e-learning and also provide computer supported collaborative learning. His current research examines improving of online collaborative learning using machine learning techniques.

Dr. Robert Oboko is a member of the faculty at the University of Nairobi’s School of Computing and Informatics. He received his Msc. Computer science from Free University of Brussels and PhD in Computer Science from University of Nairobi, Kenya in 2012. His research interests are mainly in issues around use of ICT for Development. These include issues around application of ICT in Education, Monitoring and Evaluation, health, enhancement of Social Capital, and ICT4D policy research, among others. He is also keen on the use of machine learning and mobile devices for development. He regularly publishes international refereed journal papers, refereed international conferences papers and book chapters.

Prof. Peter Waiganjo Wagacha is a faculty member at the School of Computing & Informatics, University of Nairobi. He received Bsc. Science (Hons) degree from University of Nairobi in 1991, Msc. in Computer science and Applications from Shanghai University, China in 1996 and PhD in Computer Science from University of Nairobi, Kenya in 2003. He enjoys teaching, research and working with students to develop innovative ideas and solutions. His research and extension work in ICT4D is in the areas of (1) enhancing ICT in education, such as e-learning using Artificial Intelligence, (2) mobility and urban transportation (3) human language technology for local languages (4) health informatics (5) mobile technology. He has published in refereed journals, book chapters, and conference proceedings.

How to cite this paper: Elizaphan M. Maina, Robert O. Oboko, Peter W. Waiganjo,"Using Machine Learning Techniques to Support Group Formation in an Online Collaborative Learning Environment", International Journal of Intelligent Systems and Applications (IJISA), Vol.9, No.3, pp.26-33, 2017. DOI: 10.5815/ijisa.2017.03.04