Using Self-Organizing Map (SOM) for Clustering and Visualization of New Students based on Grades

I Y Purbasari¹, E Y Puspaningrum¹ and A B S Putra²
¹Lecturer of Department of Informatics, University of Pembangunan Nasional “Veteran” Jawa Timur, Surabaya, Indonesia
²Student of Department of Informatics, University of Pembangunan Nasional “Veteran” Jawa Timur, Surabaya, Indonesia

Abstract. Student grouping, particularly in high school, is a necessary process to divide and classify students into classes based on their abilities and interests. Each school may have different approaches to decide the grouping, but most schools use academic grades. The activity occurs every new academic year and schools with plenty of new students registered may feel a bit overwhelmed with this grouping assignment. A decision support system which can automatically perform grouping on a list of students may be able to help the school’s staffs with this repetitive task. A self-organizing map (SOM) is an example of unsupervised learning algorithm using an artificial neural network structure to produce a low dimensional representation from a given input. However, SOM is also known as one of clustering techniques, since dimensionality reduction may also be seen as reducing (or clustering) input data to lower dimensions (or clusters). This research aims to group new enrolled students to a high school based on their academic grades using a SOM learning algorithm. The grades came from their rapport books and national examination results from their previous study. The resulting groups are three distinct clusters which represents Life Sciences, Social Sciences, and Linguistics study areas.

1. Introduction
In Indonesia, general high schools offer several majors (usually two or three majors) which can be chosen by future students (or in this case, may be decided by schools). Each school may have different approaches to decide the grouping, but most schools use academic grades, which is often called as ability grouping or achievement grouping. The implementation of ability grouping puts students in a place that smart children must join smart children and less smart children must join less smart children. Smart and less smart selection is done through report cards. Usually the teacher takes several top-ranking students in one class, then makes one with other students who are ranked above from other classes. This grouping aims to improve student achievement, facilitate teachers in teaching in class, make it easier for teachers to control the process of giving instructions [1]. Abilities in certain subjects are generally reflected by academic grades on those subjects. Thus, schools often group students based on their grades on that particular subjects. The activity occurs every new academic year and schools with plenty of new students registered may feel a bit overwhelmed with this grouping assignment. A decision support system which can automatically perform grouping on a list of students may be able to help the school’s staffs with this repetitive task.

A self-organizing map (SOM) is an example of unsupervised learning algorithm using an artificial neural network structure to produce a low dimensional representation from a given input. The model makes use of Kohonen Map or network [2] and is often used for dimensionality reduction so that data can be
visualized easily. Although SOM is a kind of artificial neural network (ANN), it works a bit differently than regular ANN. While in ANN applies error-correction learning, with the use of gradient descent, SOM uses competitive learning, where each node “competes” against other nodes to be the winning node, which would then get an updated weight. SOM also tries to reserve the topological structure of the input space by applying a neighbourhood function.

Artificial intelligence approach has been widely used to solve problems in every day’s life. Supervised and unsupervised learning algorithms have been developed to solve classic classification, clustering, and association problems. A Support Vector Machine has been used to classify eggs in [3] [4] [5]. A Bayesian Network Model are used for Hepatitis Diagnosis in [6] and combined with ReliefF feature selection in [7]. In [8] [9] and [10], an artificial neural network model was used to predict student’s academic achievement from early semesters input grades, and in [11] combined with Linear Regression and Support Vector Regression. In unsupervised learning, an application of SOM has been implemented to cluster students within a scholarship award scheme [12] [13]. Another application of SOM algorithm in education is to monitor e-learning activities [14] [15] and to visualize students’ cognitive structural model [16]. The use of SOM to cluster students have been investigated in [17], [18], and in [19] is used to investigate students’ interest on Maths subject.

This research aims to group new enrolled students to a high school based on their academic grades on their previous junior high schools using a SOM learning algorithm. The grades used came from their rapport books and national examination results from their previous study.

2. Self-Organizing Maps

Self-Organizing Map (SOM) is first introduced by Teuvo Kohonen in 1996. SOM implements Neural Network that aims to visualize data by reducing the dimensions of the data through use of self-organizing neural networks to map high-dimensional data into low-dimensional one. SOM applies no-guidance input-target or unsupervised learning data assuming a topology structured into clusters [20]. SOM algorithm considers the weight vector for each cluster unit as a sample from the input pattern associated with that cluster. The winning unit cluster is those whose weights match the closest input vector pattern (usually, by the square of the minimum Euclidean distance). The winning unit and its neighbouring units continue to update their weights until they conform with the input patterns. In SOM networks, target neurons are not placed in a row like other ANN models, but instead in two dimensions whose shape can be arranged. Different shapes will produce neurons around different winning neurons so the resulting weights might also be different. In SOM, weight changes are not only made to the weights of the lines connected to the winning neurons, but also to the weights of the lines to the surrounding neurons. There are various kinds of basic SOM implementation, from the shape of the map (rectangular/hexagon, 2D/3D) to the many options of the distance functions to define close neighbourhood. Figure 1 showed an architecture of a typical SOM implementation.

In its basic implementation, SOM is an algorithm that maps data in a high-dimensional vector space (input) to a two-dimensional vector space (output) located at an adjacent location, where each neuron in the input layer is connected to each neuron in the output layer. Each neuron in the output layer represents the class (cluster) of supplied input. With SOM’s nature of reducing dimensions, it is widely used as a dimensionality reduction tool, just like Principal Component Analysis (PCA) does. However, SOM is also known as one of clustering techniques, since dimensionality reduction may also be seen as reducing (or clustering) input data to lower dimensions (or clusters).

The basic steps in SOM algorithm are as follows:
1. Initialization: Determine map shape and size. Assign initial weight from input to output.
2. Sampling: Draw random samples from input adjusted to the map’s size
3. Nearest node computation: Compute closest node from map to input
4. Weight updating: Adjust weights of the neighbouring nodes of the closest node
5. Repeat step 2 to 4 until a certain iteration threshold (or a minimum threshold error value) is reached
3. Methodology

Dataset used is 275 student’s records of rapport and final examination grades from 5 subjects: Maths, Natural Sciences, English, Indonesian, and Social Sciences. There is a total number of 9 attributes (5 subjects in rapport grades and 4 subjects in final examination grades). Fig. 2 describes the flowchart of the system. The SOM algorithm was implemented using Python programming language.

3.1. Initialization

After input data was loaded, normalization was applied to range data from 0 to 1. Initialization step includes determining map’s shape and size, initialized random weights, and setting a maximum value for steps. There is various implementation of SOMs, from the shape of the map (rectangular/hexagon, 2D/3D) to the many options of the distance functions to define a “close” neighborhood. In this research, a 2D, 10 x 10 rectangular map shape was used. Each cell in map stores a 1x9-dimension data point, with initial random weights assigned. A maximum value for steps number was also set at 5,000 iterations.

3.2. Sampling

In each iteration, a random sampling of data, \( d[t] \), was drawn. The sampling was performed with replacement, so that data already selected in previous iterations, might be reselected in the later iterations.

3.3. Nearest node computation

A Best Matching Unit (BMU) was then computed which has the closest distance with the sampled input, using Euclidean Distance. The row and column numbers of the BMU value in map is returned.

3.4. Weight updating

All the neighboring nodes (in this implementation were all four adjacent nodes) of the BMU node have their weights’ updated according to the weight update rule (formula 1):

\[
    w_j(t + 1) = w_j(t) + \alpha(t)[d[t] - w_j(t)]
\]  

(1)

Where \( \alpha(t) \) is a learning rate and it decreases with time. In this implementation, the initial value of \( \alpha \) is 0.6 and in each iteration, \( s[t] \), (\( t \) ranges from 1 to 5000), it decreases by the percentage left of the iteration and computed using formula 2 and 3:
\[ p_i = \frac{1 - \left( \frac{s[i]}{m} \right)}{[i] + u} \]  
\[ u(t) = p \]  
\[ [i] = 1 - \frac{s[i]}{m} \]  
\[ [i] + u \]

Figure 2. System Flowchart

4. Result and Discussion

4.1. Result

A sample of the dataset is in Table 1. The data was normalized (in the range of 0 to 1) before being processed.

| ID   | Rapport Book | Final Exam |
|------|--------------|------------|
|      | MT | NS | EN | ID | SS | MT | NS | EN | ID |
| 180001 | 96.5 | 96.5 | 95.5 | 88.5 | 91.5 | 97.5 | 77.5 | 86 | 86 |
| 180002 | 88.5 | 88.5 | 86.5 | 87 | 89.5 | 82.5 | 90 | 74 | 88 |
| 180003 | 91 | 91 | 95 | 94.5 | 89.5 | 87.5 | 77.5 | 86 | 82 |
| 180004 | 92.5 | 92.5 | 94.5 | 93.5 | 87.5 | 87.5 | 77.5 | 82 | 86 |
| 180005 | 89.5 | 89.5 | 89.5 | 85 | 78 | 90 | 80 | 78 | 84 |
| 180006 | 91 | 91 | 94 | 92.5 | 89 | 77.5 | 85 | 80 | 88 |
| 180007 | 89.5 | 89.5 | 90.5 | 91.5 | 85 | 100 | 80 | 60 | 90 |

After map’s size is determined, which is 10x10 and each cell holds an array of 9 values, random weights were assigned. Table 2 shows a sample random weights for 1 cell:

| ID | MT | NS | EN | ID | SS | MT | NS | EN | ID |
|----|----|----|----|----|----|----|----|----|----|
| 4.170220 05e-01 | 7.203244 93e-01 | 1.143748 17e-04 | 3.023325 93e-01 | 1.467558 91e-01 | 9.233859 48e-02 | 1.862602 11e-01 | 3.455607 27e-01 | 3.967674 74e-01 |

For each iteration, BMU value is calculated and all four surrounding nodes’ weights were adjusted using formula (1). Table 3 lists the final weights for 1 cell:
Table 3. Final weights of a cell in SOM map

|        |        |        |        |        |        |        |
|--------|--------|--------|--------|--------|--------|--------|
| 0.371248 | 0.351578 | 0.374648 | 0.366969 | 0.361304 | 0.508267 | 0.594269 |
| 28     | 07     | 3      | 35     | 72     | 86     | 58     |
|        |        |        |        |        |        |        |
| 0.599300 | 0.746200 |
| 84     | 43     |        |        |        |        |        |

After 5000 iterations, the map is then visualized with U-Matrix. Unified distance matrix (U-Matrix) is used to represent a SOM visually, particularly using colours or grey scales. Adjacent nodes which are similar are visualized with the same colour indicating that they are in the same cluster. Fig. 3 is the U-matrix for the 10x10 generated SOM, where 3(a) is the U-Matrix in rainbow colours and 3(b) in greyscale.

4.2. Discussion

From the result in Fig. 3, we can see that most data fall into one large cluster: purple cluster in Fig. 3(a) or black cluster in Fig. 3(b). There are other smaller clusters on the upper left and lower left parts of the map. Overall, the map suggested that there are 2 to 3 clusters, consisting of 1 large or dominant cluster. This result corresponds to the conventional way of clustering students in high schools, which categorized study areas into 3 groups: Natural Science (IPA), Social Science (IPS), and Linguistics (Bahasa). The large cluster is usually the Natural Science cluster, which often regarded as the favourite study area.

4. Conclusions

This research has produced preliminary insight into new students’ data of grades by identifying groups of students in particular study areas. The number of potential clusters are three clusters, which adheres to the conventional grouping of Natural Science, Social Science, and Linguistics. The cluster which has the largest number of members is Natural Science cluster.

Acknowledgments

The authors would like to thank the Faculty of Computer Science Universitas Pembangunan Nasional “Veteran” Jawa Timur for its support for the publishing of this article in the conference.

References

[1] A. Imron, "Management of School-based Students" (Manajemen Peserta Didik Berbasis Sekolah), Malang: Universitas Negeri Malang, 2012.
[2] T. Kohonen, "Self-organized formation of topologically correct feature maps," Biological Cybernetics, vol. 43, no. 1, pp. 59–69, 1982.
[3] I. Y. Purbasari, F. T. Anggraeny and N. Harianto, "Classification of broiler chicken eggs using support vector machine (svm) and feature selection algorithm," in International Joint Conference on Science and Technology, Nusa Dua, 2018.
[4] M. Maimunah and T. Rokhman, "Classification of Declining Quality of Chicken Eggs Based on the Color of Shells Using Support Vector Machine "(Klasifikasi Penurunan Kualitas Telur Ayam Ras Berdasarkan Warna Kerabang Menggunakan Support Vector Machine)," Informatics for Educators and Professionals, vol. 3, no. 1, pp. 43-52, 2018.
[5] D. Nurdiyah and I. A. Muwakhid, "Comparison of Support Vector Machine and K-Nearest Neighbor for Fertile and Infertile Egg Classification Based on Glcm Texture Analysis " (Perbandingan Support Vector
Machine dan K-Nearest Neighbor Untuk Klasifikasi Telur Fertil Dan Infertil Berdasarkan Analisis Texture GLCM), *Jurnal Transformatika*, vol. 13, no. 2, pp. 29-34, 2016.

[6] S. Lakho, A. H. Jalbani, M. S. Vighio, I. A. Memon, S. S. Soomro and S. Q. N, "Decision Support System for Hepatitis Disease Diagnosis using Bayesian Network," *Journal of Computing and Mathematical Sciences*, vol. 1, no. 2, pp. 11-19, 2017.

[7] F. Anggraen, I. Purbusari and E. Suryaningsih, "ReliefF Feature Selection and Bayesian Network Model for Hepatitis Diagnosis," in *International Conferences on Information Technology and Business (ICITB)*, Bandar Lampung, 2017.

[8] F. T. Anggraen, "Prediction of Student's Academic Achievement using Artificial Neural Network (Prediksi Prestasi Akademik Mahasiswa dengan Metode Jaringan Syaraf Tiruan)," in National Seminar of Information Technology Roles in Food, Chemical, and Manufacturing Industries to Support Development (Seminar Nasional Peran Teknologi Informasi di Bidang Industri Pangan, Kimia, dan Manufaktur dalam Menunjang Pembangunan), Universitas Pembangunan Nasional "Veteran" Jawa Timur, Surabaya, 2009.

[9] S. Islamovic and M. Suknovic, "Predicting Students' Academic Performance using Artificial Neural Network: A Case Study from Faculty of Organizational Sciences," in *The Eurasia Proceedings of Educational & Social Sciences (EPESS)*, Konya, Turkey, 2014.

[10] O. L. Usman and A. O. Adenubi, "Artificial Neural Network (ANN) Model for Predicting Students' Academic Performance," *Journal of Science and Information Technology*, vol. 1, no. 2, pp. 23-37, 2013.

[11] E. Y. Obsie and S. A. Adem, "Prediction of Student Academic Performance using Neural Network, Linear Regression and Support Vector Regression: A Case Study," *International Journal of Computer Applications*, vol. 180, no. 40, pp. 39-47, 2018.

[12] L. Rahmwati, A. D. Cahyani and S. S. Putro, "Utilization of SOM-IDB cluster method as an Analysis of Scholarship Acceptance Analysis (Pemanfaatan metode cluster SOM – IDB sebagai Analisa Pengelompokan Penerimaan Beasiswa)," University of Trunojoyo Madura, Bangkalan, 2013.

[13] N. Hendayanti, G. Putri and M. Nurhidayati, "Accuracy of Classification of STMIK STIKOM Bali Scholarship Recipients with Hybrid Self Organizing Maps and K-Mean Algorithms (Ketepatan Klasifikasi Penerima Beasiswa STMIK STIKOM Bali dengan Hybrid Self Organizing Maps dan Algoritma K-Mean)," *VARIAN Journal*, vol. 2, no. 1, pp. 1-7, 2018.

[14] M. Bara, N. Ahmad, M. Modu and H. Ali, "Self-organizing map clustering method for the analysis of e-learning activities," in *Majan International Conference (MIC)*, Muscat, Oman, 2018.

[15] Y. Lee, "Using Self-Organizing Map and Clustering to Investigate Problem-Solving Patterns in the Massive Open Online Course: An Exploratory Study," *Journal of Educational Computing Research*, vol. 57, no. 2, pp. 471-490, 2019.

[16] N. Yorek, I. Ugulu and H. Aydin, "Using Self-Organizing Neural Network Map Combined with Ward's Clustering Algorithm for Visualization of Students' Cognitive Structural Models about Aliveness Concept," *Computational Intelligence and Neuroscience*, vol. 2016, p. 14pp, 2016.

[17] P. N. E. Nohuddin, Z. Zainol and A. Nordin, "Monitoring Students Performance using Self Organizing Map Trend Clustering," *International Journal of Defence Science, Engineering & Technology*, vol. 1, no. 1, pp. 50-56, 2018.

[18] R. Umar, A. Fadilil and R. Azzahra, "Self Organizing Maps (SOM) for Grouping Majors in Vocational Schools (Self Organizing Maps(SOM) untuk Pengelompokan Jurusan di SMK)," *Khazanah Informatika*, vol. 4, no. 2, pp. 131-137, 2018.

[19] S. N. Arofah and F. Marisa, "Application of Data Mining to Determine Student Interest in Mathematics Using the K-Means Clustering Method (Penerapan Data Mining untuk Mengetahui Minat Siswa pada Pelajaran Matematika menggunakan Metode K-Means Clustering)," *Journal of Information Technology and Computer Science*, vol. 3, no. 2, pp. 229-233, 2018.

[20] L. Fausett, Fundamentals of Neural Networks: Architectures, Algorithms, and Applications, New Jersey, USA: Prentice-Hall, Inc., 1994.