

Expanding Mental Outlook with the Help of Concept Maps

Burdescu Dumitru Dan, Mihăescu Marian Cristian, Ionașcu Marian Costel and Buligiu Ionut
University of Craiova, Romania
E-mail: burdescu@software.ucv.ro

Logofătu Bogdan
University of Bucharest, Romania
E-mail: logofatu@credis.ro

Keywords: mental outlook, concept maps, e-Learning, clustering

Received: October 25, 2009

Expanding mental outlook for learners is one of the important open problems in e-Learning. Online should have the property of expanding and enhancing the mental outlook of learners in general and also, in particular, concerning the studied discipline. This paper presents an approach to this issue. The tool used for expanding the mental outlook is represented by concept maps. Concept maps are used for representing relationships among concepts that define a certain area (e.g. discipline in e-Learning). The concept map is build such that it represents a concrete and broad representation of the domain. As an example, this paper presents a concept map built for Data structures and Algorithms course and more exactly for Binary Search Trees chapter.

Povzetek: Na primeru binarnih iskalnih dreves je predstavljen nov koncept e-učenja.

1 Introduction

Concept maps are a result of Novak and Gowin's [3] research into human learning and knowledge construction. Novak [1] proposed that the primary elements of knowledge are concepts and relationships between concepts are propositions. Novak [2] defined concepts as "perceived regularities in events or objects, or records of events or objects, designated by a label". Propositions consist of two or more concept labels connected by a linking relationship that forms a semantic unit. Concept maps are a graphical two-dimensional display of concepts (usually represented within boxes or circles), connected by directed arcs encoding brief relationships (linking phrases) between pairs of concepts forming propositions.

This paper uses concept maps for presenting the very higher general structure of a studied discipline. The concept map is used by learners as a top level reference material that may be consulted. This structured high level overview of the discipline is aimed to expand the mental outlook for learners in general by exercising this ability on a particular discipline.

2 Concept maps

Concept mapping may be used as a tool for understanding, collaborating, validating, and integrating curriculum content that is designed to develop specific competencies. Concept mapping, a tool originally developed to facilitate student learning by organizing key and supporting concepts into visual frameworks, can also facilitate communication among faculty and

administrators about curricular structures, complex cognitive frameworks, and competency-based learning outcomes. To validate the relationships among the competencies articulated by specialized accrediting agencies, certification boards, and professional associations, faculty may find the concept mapping tool beneficial in illustrating relationships among, approaches to, and compliance with competencies [4].

Concept maps are also effective in identifying both valid and invalid ideas held by students, and this will be discussed further in another section. They can be as effective as more time-consuming clinical interviews for identifying the relevant knowledge a learner possesses before or after instruction [7].

Recent decades have seen an increasing awareness that the adoption of refined procedures of evaluation contributes to the enhancement of the teaching/learning process. In the past, the teacher's evaluation of the pupil was expressed in the form of a final mark given on the basis of a scale of values determined both by the culture of the institution and by the subjective opinion of the examiner. This practice was rationalized by the idea that the principal function of school was selection - i.e. only the most fully equipped (outstanding) pupils were worthy of continuing their studies and going on to occupy the most important positions in society.

Ausubel [1] made the very important distinction between rote learning and meaningful learning. Meaningful learning requires three conditions: 1. The material to be learned must be conceptually clear and presented with language and examples relatable to the

learner’s prior knowledge. Concept maps can be helpful to meet this condition, both by identifying large general concepts held by the learner prior to instruction of more specific concepts, and by assisting in the sequencing of learning tasks through progressively more explicit knowledge that can be anchored into developing conceptual frameworks; 2. The learner must possess relevant prior knowledge. This condition can be met after age 3 for virtually any domain of subject matter, but it is necessary to be careful and explicit in building concept frameworks if one hopes to present detailed specific knowledge in any field in subsequent lessons. We see, therefore, that conditions (1) and (2) are interrelated and both are important; 3. The learner must choose to learn meaningfully. The one condition over which the teacher or mentor has only indirect control is the motivation of students to choose to learn by attempting to incorporate new meanings into their prior knowledge, rather than simply memorizing concept definitions or propositional statements or computational procedures. The indirect control over this choice is primarily in instructional strategies used and the evaluation strategies used. Instructional strategies that emphasize relating new knowledge to the learner’s existing knowledge foster meaningful learning. Evaluation strategies that encourage learners to relate ideas they possess with new ideas also encourage meaningful learning. Typical objective tests seldom require more than rote learning [9].

According to this approach, the responsibility for failure at school was to be attributed exclusively to the innate (and, therefore, unalterable) intellectual capacities of the pupil. The learning/ teaching process was, then, looked upon in a simplistic, linear way: the teacher transmits (and is the repository of) knowledge, while the learner is required to comply with the teacher and store the ideas being imparted. [5]. Usage of concept maps may be very useful for students when starting to learn about a subject. The concept map may bring valuable general overlook of the subject for the whole period of study. It may be advisable that at the very first meeting of students with the subject to include a concept map of the subject.

3 Data structures and algorithms / Binary search tree concept map

The experimental concept map was used on Tesys e-Learning platform [6]. On this platform there was set an Algorithms and Data Structures discipline. The tests were performed for five chapters: Simply/Double Linked Lists, Binary Search Trees, Height Balanced Trees, B Trees and Graphs.

The concepts are presented in table 1.

ID	Concept	ID	Concept
C1	BST	C9	Right child
C2	Dynamic Structure	C10	No child
C3	Node(s)	C11	Root
C4	Traversed	C12	Leaf
C5	Key	C13	Preorder
C6	Parent	C14	Inorder
C7	Child	C15	Postorder

C8	Left child	C16	Ascending order
----	------------	-----	-----------------

Table 1: List of concepts

The concept map for Binary Search Trees is presented in figure 1. It contains 16 concepts, 11 linking phrases and 16 propositions.

The list of propositions with two concepts and one linking phrase is presented in table 2. The list of propositions with three concepts and two linking phrases is presented in table 3.

Once the concept map has been built the general graph of the each chapter may be created. In this graph, each proposition will become an edge that links the first concept and the last concept. The domain knowledge expert will assign a weight for each edge. While the students answers questions the number of correct and wrong answers will determine the knowledge weight of that edge.

There is one proposition with five concepts and four linking phrases: “BST” may be “Traversed” in “Preorder” determines “Key” in “Ascending Order”. The concepts are bolded and put between quotation marks, while linking phrases are italic and underlined.

Id	Concept	Linking phrase	Concept
P1	BST	is	Dynamic Structure
P2	BST	is made of	Node(s)
P3	Node	has	key
P4	Node	is	Parent
P5	Node	is	Child
P6	Parent	is greater than	Left child
P7	Parent	is smaller than	Right child
P8	Node	may have	Left child
P9	Node	may have	Right child
P10	Node	may have	No child

Table 2: List of propositions with two concepts and one linking phrase.

Id	C	LP	C	LP	C
P11	Node	without	parent	is	root
P12	BST	may be	traversed	in	preorder
P13	BST	may be	traversed	in	inorder
P14	BST	may be	traversed	in	postorder

Table 3: List of propositions with three concepts and two linking phrases.

Once the Concept Map has been set up the professor has to set up the quiz questions for the chapter. For each edge in the map it will correspond a certain number of quiz questions. There is no specification regarding the number of quiz questions but a minimum (e.g. five) number is still required. The activity performed by learner is monitored and stored. The Calculus engine will reconstruct an Annotated Concept Map which will present to the learner the current status of his knowledge level at Concept level. In this way, the learner will have an exact outlook of his knowledge level regarding that chapter. This information may be used by learners and course managers to obtain important information regarding the reached knowledge level.

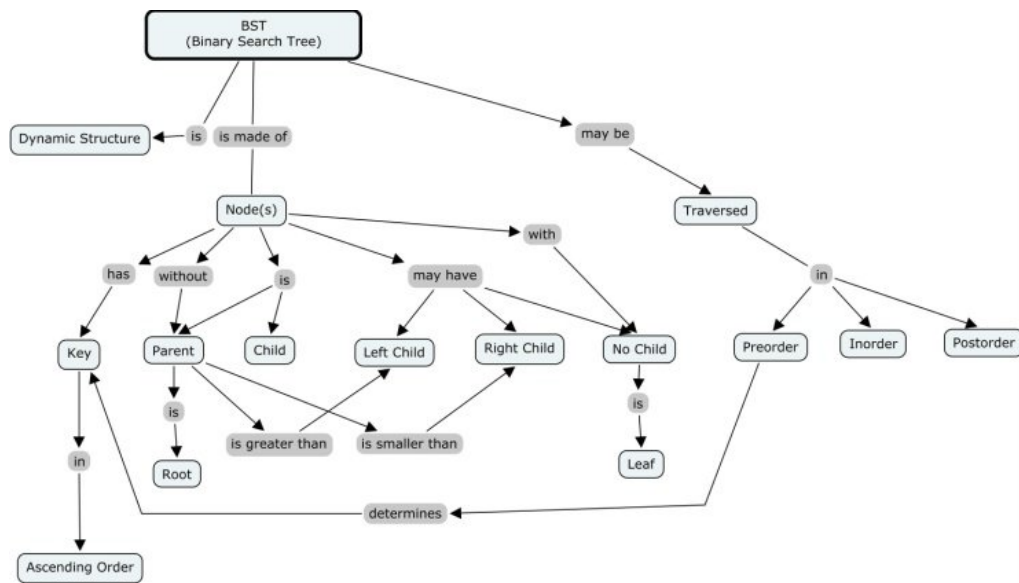


Figure 1: Binary Search Tree Concept Map

Defining a metric for computing the accumulated knowledge is accomplished. If

- W** is the weight of the edge,
- CA** is the number of correct answers,
- WA** is the number of wrong answers,
- N** is the number of questions

Then

KW is the knowledge weight of the edge

$$KW = \frac{CA - WA}{N} \frac{1}{W} * 100$$

Under these circumstances the knowledge weight may also be negative. At any time there may be estimated the overall knowledge level of the learner as the ratio between overall knowledge weight and overall weight.

Proposition	Weight	Number of questions
P1	10	8
P2	4	7
P3	7	6
P4	3	5
P5	2	7

Table 4: Sample setup of BST chapter.

Proposition (weight)	No. of questions	CA	WA	KW (%)
P1 (10)	8	3	2	1.25
P2 (4)	7	4	2	7.14
P3 (7)	6	1	3	-4.76
P4 (3)	5	3	1	13.3

Table 5: Sample values for learner’s associated graph.

Chapter	Weight
Simply/Double Linked Lists	15
Binary Search Trees	15
Height Balanced Trees	25
B Trees	25
Graphs	20

Table 6: Sample weights assigned to chapters.

Table 4 presents a sample of the setup of the Binary SearchTrees chapter. Table 5 presents a sample of the of the values of the Learner’s Associated Graph corresponding to BST chapter.

The values from table five are marked in an Annotated Concept Map that is finally presented to the learner. The Annotated Concept Map is the final outcome of the Decision Support System and is supposed to guide the learner regarding the necessary future efforts.

Table 6 presents the weights of chapters as they were assigned by the domain expert.

4 General infrastructure

For running such a process the e-Learning infrastructure must have some characteristics. The process is designed to run at chapter level. This means a discipline needs to be partitioned into chapters. The chapter has to have assigned a concept map which may consist of about 20 concepts. Each concept has assigned a document and a set of quiz questions. Each concept and each quiz has a weight, depending of its importance in the hierarchy. It is supposed that the platform run for some time such that there is a history of performed activities for a large number of learners. Figure 2 presents a general e-Learning infrastructure for a discipline.

Once a course manager has been assigned a discipline, he has to set up its chapters by specifying their names and their associated concept maps. For each concept managers have the possibility of setting up one document and one pool of questions. When the discipline

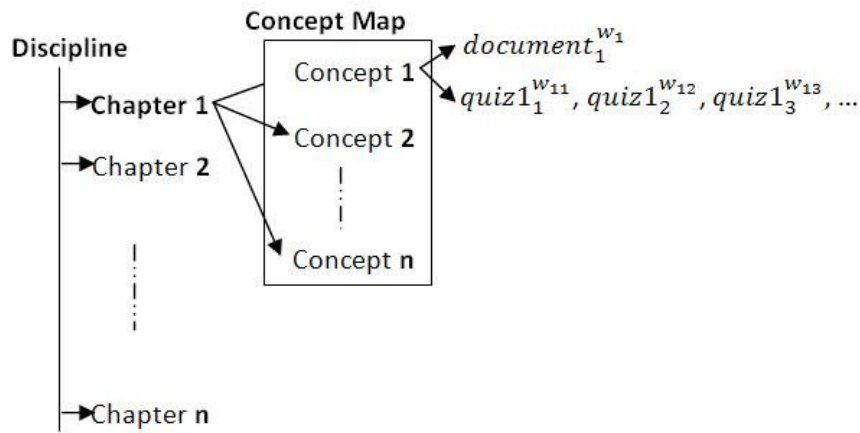


Figure 2: General structure of a discipline

is fully set, the learning process may start for learners. Any opening of the document and any test quiz that is taken by a learner is registered. The business logic of document retrieval tool wills using these data in order to determine the moment when it is able to determine the document (or the documents) that are considered to need more attention from the learner. The course manager specifies the number of questions that will be randomly extracted for creating a test or an exam.

Let us suppose that for a chapter the professor created 50 test quizzes and he has set to 5 the number of quizzes that are randomly withdrawn for testing and 15 the number of quizzes that are randomly withdrawn for final exam. It means that when a student takes a test from this chapter 5 questions from the pool of test question are randomly withdrawn. When the student takes the final examination at the discipline from which the chapter is part, 15 questions are randomly withdrawn. This manner of creating tests and exams is intended to be flexible enough for the professor.

5 Experimental results

The setup and experimental results were obtained on Tesys e-Learning platform [6]. On this platform there was set Algorithms and Data Structures discipline. The tests were performed for the chapter named Binary Search Trees. Figure 1 presents the concept map for Binary Search Trees chapter.

For each concept there is associated a set of quizzes. As learners start answering quizzes the table data regarding performed activities is continuously updated. Table 1 presents the structure of recorded activities. Table 2 presents a snapshot for the activities table. The total number of students of 400 and the total number of problems is 200. The time during which results have been taken is six months.

In order to obtain relevant results we pruned noisy data. We considered that students for which the number of taken tests or the time spent for testing is close to zero are not interesting for our study and degrade performance and that is why all such records were deleted. After this step there remained only 268 instances.

Event type	Details
login	start time and duration
view document	start time, duration, concept
self test	start time, duration, quizzes
messages	start time, duration, receiver categories

Table 7: Structure of recorded activities

User id	Event type	Details
10	login	10.10.2009 11:20, 25 min
11	view document	10.10.2009 11:21, 10 min, concept11
12	self test	10.10.2009 11:31, 3 min, concept11
13	messages	10.10.2009 11:34, 5 min, prof3, prof4

Table 8: Sample data from activities table

After data has been recorded, there was run a clustering process on learners. Clustering is the process of grouping a set of physical or abstract objects into classes of similar objects. Basically, for our platform we create clusters of learners based on their activity. There are many clustering methods in the literature: partitioning methods, hierarchical methods, density-based methods, grid-based methods or model-based methods. From all of these we chose to have a closer look on partitioning methods.

Given a database of n objects and k , the number of clusters to form, a partitioning algorithm organizes the objects into k partitions ($k \leq n$), where each partition represents a cluster. The clusters are formed to optimize an objective partitioning criterion, often called similarity function, such as distance, so that objects within a cluster are “similar”, whereas the objects of different clusters are “dissimilar” in terms of database attributes. So, the first step is to define a list of attributes that may be representative for modelling and characterizing student’s activity.

Among the attributes there may be:

- the number of logins,

- the number of taken tests,
- the average grade for taken tests

The classic k-means algorithm is a very simple method of creating clusters. Firstly, it is specified how many clusters are being thought: this is the parameter k. Then k points are chosen at random as cluster centres. Instances are assigned to their closest cluster centre according to the ordinary Euclidean function. Next the centroid, or the mean, of all instances in each cluster is calculated – this is the “means” part. These centroids are taken to be the new centre values for their respective clusters. Finally, the whole process is repeated with the new cluster centres. Iteration continues until the same points are assigned to each cluster in consecutive rounds, at each point the cluster centres have stabilized and will remain the same thereafter.

From a different perspective for a cluster there may be computed the following parameters:

- $\mu = \frac{x_1 + x_2 + \dots + x_n}{n}$, the means
- $\sigma = \frac{(x_1 - \mu)^2 + (x_2 - \mu)^2 + \dots + (x_n - \mu)^2}{n - 1}$, the

standard deviation

- p, the probability

The sum of all probabilities for all clusters is 1. If we know which of the distributions each instance came from, finding the parameters is easy. On the other hand, if the parameters are known finding the probabilities that a given instance comes from each distribution is easy. Given an instance x, the probability that it belongs to cluster A is:

$$\Pr[A|x] = \frac{\Pr[x|A] - \Pr[A]}{\Pr[x]} = \frac{f(x; \mu_A, \sigma_A) p_A}{\Pr[x]}$$

where $f(x; \mu_A, \sigma_A)$ is the normal distribution function for cluster A, that is:

$$f(x; \mu_A, \sigma_A) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

The EM algorithm takes into consideration that we know neither of these things: not the distribution that each training instance came from, nor the parameters μ , σ or the probability. So, we adopt the procedure used for the k-means clustering algorithm and iterate. Start with initial guess for the five parameters, use them to calculate the cluster probabilities for each instance, use these probabilities to reestimate the parameters, and repeat. This is called the EM algorithm for “expectation-maximization”. The first step, the calculation of cluster probabilities (which are the “expected” class values) is “expectation”; the second, calculation of the distribution parameters is “maximization” of the likelihood of the distributions given the data.

A slight adjustment must be made to the parameter estimation equations to account for the fact that it is only cluster probabilities, not the clusters themselves, that are known for each instance. These probabilities just act like

weights. If w_i is the probability that instance i belongs to cluster A, the mean and standard deviation are:

$$\mu_A = \frac{w_1 x_1 + w_2 x_2 + \dots + w_n x_n}{w_1 + w_2 + \dots + w_n}$$

$$\sigma_A^2 = \frac{w_1 (x_1 - \mu)^2 + w_2 (x_2 - \mu)^2 + \dots + w_n (x_n - \mu)^2}{w_1 + w_2 + \dots + w_n}$$

where x_i are all the instances, not just those belonging to cluster A. Technically speaking, this is a “maximum likelihood” estimator of the variance.

Now we shall discuss about how iterating process is terminated. The k-means algorithm stops when the classes of instances don’t change from one iteration to the next – a “fixed point” has been reached. In the EM algorithm things are not quite so easy: the algorithm converges toward a fixed point but never actually gets there. But we can see how close it is by calculating the overall likelihood that the data came from this dataset, given the values for the parameters (mean, standard deviation and probability). This overall likelihood is obtained by multiplying the probabilities of the individual instances i:

$$\prod_i (p_A \Pr[x_i | A] + p_B \Pr[x_i | B])$$

Where the probabilities given the clusters A and B are determined from the normal distribution function $f(x; \mu, \sigma)$. This overall likelihood is a measure of the “goodness” of clustering and increases at each iteration of the EM algorithm. Again, there is a technical difficulty with equating the probability of a particular value of x with $f(x; \mu, \sigma)$, and in this case the effect does not disappear because no probability normalization operation is applied. The upshot is that the likelihood expression above is not a probability and does not necessarily lie between zero and one: nevertheless, its magnitude still reflects the quality of the clustering. In practical implementations its logarithm is calculated instead: this is done by summing the logs of individual components, avoiding all the multiplications. But the overall conclusion still holds: iterate until the increase in log-likelihood becomes negligible. For example, a practical implementation might iterate until the difference between successive values of log-likelihood is less than 10⁻¹⁰ for ten successive iterations. Typically, the log-likelihood will increase very sharply over the first few iterations and then converge rather quickly to a point that is virtually stationary.

Although the EM algorithm is guaranteed to converge to a maximum, this is a local maximum and may not necessarily be the same as the global maximum. For a better chance of obtaining the global maximum, the whole procedure should be repeated several times, with different initial guess for the parameter values. The overall log-likelihood figure can be used to compare the different final configuration obtained: just choose the largest of the local maxima.

The EM algorithm is implemented in Weka package [10] and needs the input data to be in a custom format

called *arff*. Under these circumstances we have developed an offline Java application that queries the platform's database and crates the input data file called *activity.arff*. This process is automated and is driven by a *properties* file in which there is specified what data will lay in *activity.arff* file.

The most important step in this procedure is the attribute selection and the granularity of their nominal values. The number of attributes and their meaning has a crucial importance for the whole process since irrelevant attributes may degrade classification performance in sense of relevance. On the other hand, the more attributes we have the more time the algorithm will take to produce a result. Domain knowledge and of course common sense are crucial assets for obtaining relevant results.

Running the EM algorithm created four clusters. The procedure clustered 40 instances (15%) in cluster 0, 83 instances (31%) in cluster 1, 112 instances (42%) in cluster 2 and 33 instances (12%) in cluster 3. The final step is to check how well the model fits the data by computing the likelihood of a set of test data given the model. Weka measures goodness-of-fit by the logarithm of the likelihood, or log-likelihood: and the larger this quantity, the better the model fits the data. Instead of using a single test set, it is also possible to compute a cross validation estimate of the log-likelihood. For our instances the value of the log-likelihood is -3.75 which represents a promising result in the sense that instances (in our case students) may be classified in three disjoint clusters based on their activity.

6 Conclusions and future work

Tesys e-Learning platform has been designed such that on-line testing activities may be performed as they were set up by course managers.

A Concept Map for a Binary Search Trees chapter as well as for each chapter of Algorithms and Data Structures course has been created. The Concept maps have been the starting point in creating the sets of quiz questions. Each quiz question refers to a certain proposition from the concept map.

After the setup has been put in place, the learners started using the platform. At request, from the general concept map there was derived the learner's associated concept map. This concept map provides important information regarding the level of knowledge reached by learner.

The business logic computes the knowledge of the student regarding the chapter as a knowledge weight and regarding the discipline as a percentage of covered concepts. This weight is computed as a function of proposition's weight, number of questions assigned to that proposition, the number of correct answered questions and number of wrong answered questions.

We have created a procedure of data analysis which may provide interesting conclusions regarding the classification of students from an e-learning platform.

The platform was developed, it is currently running and has built in capabilities of monitoring students testing activities. An off-line application was developed

for creating the input data files that are analysed. Data analysis is done using EM clustering algorithm implemented by Weka system. One of the main goals is clustering of students. This may lead to important information regarding the learners for which expanding mental outlook performed better.

Student's clustering may have a predictive value in the sense that from the performed testing activities a student has made there may be pulled conclusions about his learning proficiency. On the other hand, platform's characterization may have as result an estimation of the capability of an e-learning system to grade and order students according to their accumulated knowledge. This analysis is critical for having as conclusion that a system can expand mental outlook.

This whole mechanism represents the functionality of a decision support system that runs along the Tesys e-Learning platform.

Whenever needed, the learner may study his own associated concept map at discipline level or at chapter level. This functionality has as main benefit expanding the mental outlook in general and for studied discipline in particular.

As future works, a computational framework may be created which estimates the proficiency of the learners after using the concept map as structuring facility. This kind of analysis may give important information concerning the improvement of the used concept map.

References

- [1] Novak, J. D. (1977). *A Theory of Education*. Cornell University Press, Ithaca, NY.
- [2] Novak, J. D. (1998). *Learning, Creating, and Using Knowledge: Concept Maps as Facilitative Tools in Schools and Corporations*. Lawrence Erlbaum Associates, Mahwah, NJ.
- [3] Novak, J. D. and Gowin, D. B. (1984). *Learning How to Learn*. Cambridge University Press, New York.
- [4] McDaniel, E., Roth, B., and Miller, M. (2007) Concept Mapping as a Tool for Curriculum Design. *Issues in Informing Science and Information Technology*, Vol. 4.
- [5] Vecchia, L., Pedroni, M. (2007). Concept Maps as a Learning Assessment Tool. *Issues in Informing Science and Information Technology*, Vol. 4.
- [6] Burdescu, D.D., Mihăescu, M.C., 2006. Tesys: e-Learning Application Built on a Web Platform. In *Proceedings of International Joint Conference on e-Business and Telecommunications*. INSTICC Press, Setubal, Portugal, pp. 315-318
- [7] Edwards, J. And Fraser, K. (1983). Concept maps as reflections of conceptual understanding. *Research in Science Education*, 13, pp. 19-26.
- [8] Ausubel, D. P., Novak, J. D., and Hanesian, H., (1978). *Educational Psychology: A Cognitive View (2nd ed.)*. Holt, Rinehart and Winston, New York.
- [9] Holden, C. (1992). Study flunks science and math tests. *Science Education*, 26, 541.
- [10] WEKA, www.cs.waikato.ac.nz/ml/weka