Paving the way for providing teaching feedback in automatic evaluation of open response assignments

Verónica Bolón-Canedo*, Jorge Díez†, Oscar Luaces†, Antonio Bahamonde†,Amparo Alonso-Betanzos*
*Laboratory for Research and Development in Artificial Intelligence (LIDIA), Computer Science Dept.,
University of A Coruña, 15071 A Coruña, Spain
{vbolon, ciamparo}@udc.es
†Artificial Intelligence Center. University of Oviedo, Gijón, Spain
{jdiez, oluaces, abahamonde}@uniovi.es

Abstract—**Peer grading has been the regular procedure to use for automatic assessment of open ended assignments in Massive Open Online Courses (MOOCs). However, and although the procedure tries to overcome the rupture of the classical teach-learn-assess/feedback cycle, it does so only in the student side, and no attempt has been made as yet in giving feedback to instructors. The work described inhere aims at filling this gap, with a proposal in which the instructors are supplied with the set of words most used by the best and worst ranked quartiles of assignments. In order to achieve this, a Gaussian Mixture Model (GMM) fed with the *bag of words* supplied by a previous feature selection algorithm is presented, with the goal of identifying the clusters of words related with similar grades. The results obtained over three pilot studies, containing assignments in three different disciplines, show that our model can lead to more complete information on the teacher feedback on the results of the assignments.**

I. Introduction

Massive Open Online Courses (MOOCs) are a new educational format, that was born around in the 2000s, and that since then has been progressively increasing in number of courses offered. Perhaps the reason for such a successful history is that this type of instruction gives response to a society requirement on lifelong learning, that demands flexibility and compatibility with regular professional activities, which in traditional education are not feasible. Thus, education tends to a more personalized but at the same time, economically and temporally restricted scenario. There are basic key differences between MOOCs and traditional instructional modes, being scale one of those critical differences, with MOOCs having tens of thousands of students enrolled. This fact originates that the classical teach-learn-assess/feedback cycle is broken [1]. But the formative assessment/feedback step, both on the students and the instructors sides, is critical to guide subsequent instruction and ensure learning. Thus, two different problems are to be addressed, one is to provide assessment on their learning results to the students, and another different one is to give feedback to instructors on how students are managing the learned concepts, beyond the qualification received.

On the assessment of the students, several attempts have been made to re-introduce some degree of formative feedback into the process, to prevent it from becoming a one-way information broadcasting situation. Many methods are suitable for feedback in an open distance learning environment [2], but few are applicable to MOOCs scale, being peer assessment one of the most widely employed, as it is applicable to all types of contents and assignments, and also prevents the need to have a large pool of support instructional tutors. In this type of assessment, fellow students within a MOOC are asked to evaluate student assignments and to provide feedback to other students.

While peer assessment has been the focus of attention of several research works [3], [4], [5], feedback to instructors on the results obtained by students beyond their qualification has been mostly ignored in scientific literature, despite being an important aspect to restore the teach-learn-assess cycle. Providing teachers with comprehensive material than can help them to improve teaching methods and materials is a complicated task in peer assessment, as up to now, the only result available for them is the qualification obtained by the students, over which some statistical measures could be obtained.

In this work, our aim is to analyze the use that the students make of the corpus of words in their answers to the assessments, relating them to the qualifications obtained in the peer assessment process. The idea behind the methodology is to be able to obtain clusters of words that are used by the best and the worst clusters of responses, thus providing the instructor with a representation of what has been learned by the students, giving the former the opportunity to reshape materials that can guide the students to better achieve the learning goal. Clustering words for other applications has been addressed previously by [6], in which it is described a method for clustering words automatically according to their distribution in particular syntactic contexts. Clusters are obtained with the lowest possible distortion using deterministic annealing, although their aim is completely different from the one described inhere, as clusters are used as the basis for class models of word co-occurrence with predictive power (for example, for establishing missing words). Based on this later work, in [7] a probabilistic algorithm for clustering words in a document classification context is provided. The words are similar if class distributions are, using the Kullback–Leibler distance between distributions. Finally, in [8] the same idea, but employing a more temporal efficient algorithm is devised. Different from these previous works, our goal is to restore in some way, the teach-learn-feedback cycle that is broken, in this particular proposal because instructors do not evaluate all the assessments, and thus provide them with an idea of the use that students make of the corpus of words of the subject studied. In our work, the aim is to find clusters of words that relate with similar grades using a Gaussian Mixture Model (GMM) [9]. GMMs are composed of k multivariate normal density components, being k a positive integer. Each component has a d-dimensional mean (being d a positive integer), d-by-d covariance matrix, and a mixing proportion. The mixing proportion j determines the proportion of the population composed by component j where $j = 1, ..., k$. Once the GMM is fitted, it can be used to cluster data. The posterior probabilities for each point indicate that each data point has some probability of belonging to each cluster. In this work we initialize the GMM over those words that were previously chosen by using a feature selection ranker. Such clusters are then used to help the instructors in identifying which concepts and which not have been correctly learned by students. The general process is the following:

1) The input data is a set of grades given by student graders over a set of other students' answers. These data constitute the training dataset for our learning algorithm.
2) The words used in the answers and the graders are both represented in a common Euclidean space. The dot product between both representations will be the learned qualification, that should be coherent with the data of the real qualifications. Thus, the location of the elements has a semantic meaning: near points will mean near qualifications. As both are on an Euclidean space, the distances (that measure semantic alikeness) will be calculated by means of the euclidean distances.
3) Using a Gaussian Mixture Model (GMM) for studying the distribution of the representation of the words, clusters of words that are significantly near are to be searched for.
4) To fit the GMM, words that have a meaningful role in the process are to be used. As words play the role of variables in the learning of the grades assigned, variable selection techniques will be employed. Specifically, feature selection rankers are used to obtain an ordered list of words, from which the top ones are chosen by means of a threshold. This is a fundamental step in the process, so as to obtain the set of most representative words, that will later be used for the GMM model, making it feasible, as will be shown in section III.
5) Finally, the instructors are supplied with the words that conform the clusters obtained.

There have been, to our knowledge, no previous attempts to provide feedback to instructors in the literature. In our approach this is achieved using a representation of the words employed in a euclidean space with semantic implications, and feature selection methods to restrict the words appearing in the clusters devised.

II. Formal Framework

Let $\mathcal{G}$ be a set of *Graders* and let $\mathcal{A}$ be a set of *Answers*. Each grader g has received a subset $\mathcal{A}_g \subset \mathcal{A}$ of answers to evaluate. The initial data to infer a grading function is the *assessment matrix*, $\boldsymbol{M}$, which contains the scores given by the graders:

$$\boldsymbol{M}(\boldsymbol{g}, \boldsymbol{a}) \in [0, 10], \, \boldsymbol{g} \in \mathcal{G}, \, \boldsymbol{a} \in \mathcal{A}. \tag{1}$$

In general, this matrix is going to be very sparse. Only a few answers will be evaluated by each student. The goal of any peer-assessment method is to obtain an absolute ranking of answers from the scores in $\boldsymbol{M}$.

Both graders and answers will be represented by vectors of features; we will use the same symbols to name their vectorial representation or the grader or answer. In the simplest case, a grader can be identified by a vector of binary values with all zeros but one 1 in the component indexed by itself in $\mathcal{G}$. However, this simple representation can be enriched with features describing additional aspects of the graders. For example, the representation might include demographic data of the student: age, gender, or historical data based on previous peer-assessments.

On the other hand, the answers can be represented by their words. Although many other codifications can be utilized, we used a shallow natural language processing. Thus we borrow from Information Retrieval the term-document matrix, $\boldsymbol{T}$, which represents the occurrence of terms (in columns) in a set of documents (in rows), in this case answers. As usual, we remove form the *corpus* a list of stop-words, but we did not use any stemming procedure. Then we represented each answer by the corresponding row in $\boldsymbol{T}$. This is usually known as *Vector Space Model* (VSM) or term vector model [10], [11].

If a term w occurs in a document (answer), its value in the corresponding column is non-zero. Although several different ways of computing these values– also known as (term) weights– have been developed, we will consider the simplest one: that is, the weight will be 1 if the term appears, 0 if it does not. The definition of term depends on the application. Typically terms are single words, keywords, or longer phrases. If words are chosen to be the terms, the dimensionality of the vector is the number of words in the vocabulary (the number of distinct words occurring in the corpus) excluding the stop-words. In our approach, first we learn a scoring function able to fill the matrix $\boldsymbol{M}$ and then use the average scores of all graders on all answers to obtain the final ranking. This scoring function is induced based on preference learning to avoid the subjectivity of graders. The focus is on the relative ordering of answers for each grader, and not in the score values. Thus, a set of *preference judgments*, $\mathcal{D}$, is built, given by triples of a grader $\boldsymbol{g}$ and a couple of answers $(\boldsymbol{a}_b, \boldsymbol{a}_w)$ (where $\boldsymbol{a}_b$ is a better assessed answer than $\boldsymbol{a}_W$) in $\mathcal{A}_g$, such that

$$\boldsymbol{M}(\boldsymbol{g}, \boldsymbol{a}_b) > \boldsymbol{M}(\boldsymbol{g}, \boldsymbol{a}_w) \Rightarrow (\boldsymbol{g}, \boldsymbol{a}_b, \boldsymbol{a}_w) \in \mathcal{D}. \tag{2}$$

Notice that answers with the same score will not provide any relative order, so ties are discarded when generating the dataset of preference judgments.

The strategy that we propose to obtain the ranking starts with a double *embedding*: mapping both answers and graders into a common Euclidean space $\mathbb{R}^k$ for some integer k,

$$\mathbb{R}^{|\mathcal{G}|} \to \mathbb{R}^k, \quad \boldsymbol{g} \mapsto \boldsymbol{W}\boldsymbol{g}; \tag{3}$$
$$\mathbb{R}^{|rep(\mathcal{A})|} \to \mathbb{R}^k, \quad \boldsymbol{a} \mapsto \boldsymbol{V}\boldsymbol{a}. \tag{4}$$

From dataset $\mathcal{D}$ and with the embeddings, we will define the *individual assessment* as an *utility* function from graders and answers as follows.

$$u(\boldsymbol{g}, \boldsymbol{a}) = \langle \boldsymbol{W}\boldsymbol{g}, \boldsymbol{V}\boldsymbol{a} \rangle \tag{5}$$

such that close objects in $\mathbb{R}^k$ will have similar properties; i.e. neighboring graders will evaluate similarly a given answer, and neighboring answers will get a similar score for a fixed grader. This function estimates the grade given by any grader $\boldsymbol{g}$ to any answer $\boldsymbol{a}$. However, in order to fill the assessment matrix, we can compute the *final grade* for each answer as the average of all its grades:

$$\frac{1}{|\mathcal{G}|}\sum_{g\in\mathcal{G}} u(\boldsymbol{g}, \boldsymbol{a}) = \frac{1}{|\mathcal{G}|}\sum_{g\in\mathcal{G}} \langle \boldsymbol{W}\boldsymbol{g}, \boldsymbol{V}\boldsymbol{a} \rangle =$$
$$\Big\langle \frac{1}{|\mathcal{G}|}\sum_{g\in\mathcal{G}} \boldsymbol{W}\boldsymbol{g}, \boldsymbol{V}\boldsymbol{a} \Big\rangle = \langle \boldsymbol{W}\bar{\boldsymbol{g}}, \boldsymbol{V}\boldsymbol{a} \rangle = u(\bar{\boldsymbol{g}}, \boldsymbol{a}), \tag{6}$$

where $\bar{\boldsymbol{g}}$ is a vector representing the *average grader*,

$$\bar{\boldsymbol{g}} = \frac{1}{|\mathcal{G}|}\sum_{\boldsymbol{g}\in\mathcal{G}} \boldsymbol{g}.$$

In order to consider at the same time individual and final grades, we are trying to find the embedding matrices $\boldsymbol{W}$, (Eq. 3), and $\boldsymbol{V}$, (Eq. 4), that give rise to most similar ranking with those provided by graders. In a sense that we are going to explain next, we optimize the function

$$u(\bar{\boldsymbol{g}}, \boldsymbol{a}) + u(\boldsymbol{g}, \boldsymbol{a}) = \langle \boldsymbol{W}\bar{\boldsymbol{g}}, \boldsymbol{V}\boldsymbol{a} \rangle + \langle \boldsymbol{W}\boldsymbol{g}, \boldsymbol{V}\boldsymbol{a} \rangle =$$
$$\langle \boldsymbol{W}(\bar{\boldsymbol{g}} + \boldsymbol{g}), \boldsymbol{V}\boldsymbol{a} \rangle = u(\bar{\boldsymbol{g}} + \boldsymbol{g}, \boldsymbol{a}). \tag{7}$$

Now, let us establish how to make the comparison of two rankings. We are going to compute the proportion of pairs of answers whose relative order is the same. That is to say, we use the area under the ROC curve (AUC). This metric is also known as the *concordance index* (C-index), the pairwise ranking accuracy, or the *Kendall-τ*. In symbols, the similarity of a grading function h and the ranking registered in $\mathcal{D}$ is given by

$$\mathrm{AUC}(h, \mathcal{D}) = \frac{1}{|\mathcal{D}|}\sum_{(\boldsymbol{g}, \boldsymbol{a}_b, \boldsymbol{a}_w)\in\mathcal{D}} \mathrm{Score}(h, \boldsymbol{g}, \boldsymbol{a}_b, \boldsymbol{a}_w), \tag{8}$$
$$\mathrm{Score}(h, \boldsymbol{g}, \boldsymbol{a}_b, \boldsymbol{a}_w) = \mathbb{I}_{h(\boldsymbol{g}, \boldsymbol{a}_b) > h(\boldsymbol{g}, \boldsymbol{a}_w)} + \frac{1}{2}\mathbb{I}_{h(\boldsymbol{g}, \boldsymbol{a}_b) = h(\boldsymbol{g}, \boldsymbol{a}_w)}.$$

This measure is not symmetric, so when comparing two rankings we have to explicitly consider one of them as the *ground truth* and the other as the predicted ranking. In (8) we evaluate the quality of the ranking induced by h considering that the preference judgments in $\mathcal{D}$ represent the true ranking.

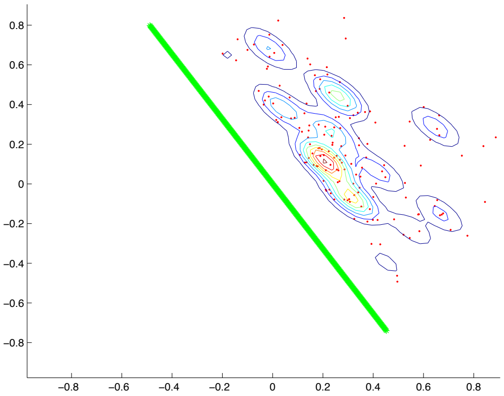


Fig. 1. Map of words involved in the answers of an assignment and fitted Gaussian Mixture contours

Tying up all the loose ends, the aim of the learning process devised to make the assessment is to optimize the embedding matrices in such a way that the individual plus the final grades be as coherent with graders' orderings as possible. Since the AUC (8) is not a convex function, we will follow a maximum margin approach. Then, we define

$$\mathrm{err}(\boldsymbol{W}, \boldsymbol{V}) = \tag{9}$$
$$\sum_{(\boldsymbol{g}, \boldsymbol{a}_b, \boldsymbol{a}_w)\in\mathcal{D}} \max\Big\{0, 1 - u(\bar{\boldsymbol{g}} + \boldsymbol{g}, \boldsymbol{a}_b) + u(\bar{\boldsymbol{g}} + \boldsymbol{g}, \boldsymbol{a}_w)\Big\}.$$

The idea is to ensure that the difference of sum of individual and final grades estimated for $\boldsymbol{a}_b$ and $\boldsymbol{a}_w$ is at least 1. To learn the parameters that minimize the previous equation we may use a Stochastic Gradient Descent (SGD) algorithm. For a more detailed description of the framework, please refer to [3].

III. Sparse Solution using Feature Selection Filters

Feature selection (FS) is a machine learning discipline that has proved to be successful in a wide number of applications. FS has been used as a preprocessing step to reduce the number of words to be used in a text classification or prediction task [12], [13], by decreasing the size of effective vocabulary, obtaining a vector space or *bag of words* model. Most applications perform FS to select the words that will be subsequently fed to a learning algorithm, both for accuracy and scalability reasons.

In our context, the final grade (6) of an answer can be expressed as the sum of grades of its words.

$$u(\bar{\boldsymbol{g}}, \boldsymbol{a}) = \langle \boldsymbol{W}\bar{\boldsymbol{g}}, \boldsymbol{V}\boldsymbol{a} \rangle = \sum_{\boldsymbol{w}\in\boldsymbol{a}} \langle \boldsymbol{W}\bar{\boldsymbol{g}}, \boldsymbol{V}\boldsymbol{w} \rangle, \tag{10}$$

where $\boldsymbol{w}$ is the vector that represents the answer with only the word w. The vectorial representation of this word has only one 1 and the rest of components are 0. Therefore, $\boldsymbol{V}\boldsymbol{w}$ is the w-th column of matrix $\boldsymbol{V}$: a point in $\mathbb{R}^k$, see Figure 1.

Therefore, the words of the corpus are the features involved in the assessment of answers in open response answers. Let us recall that there are a few very common words, and many words that rarely appear. Thus, the most predictive features would be those that appear frequently in one class, but not in the other. Additionally, remember that we have excluded the stop-words that, in fact, are low quality features since they are present in most of the answers.

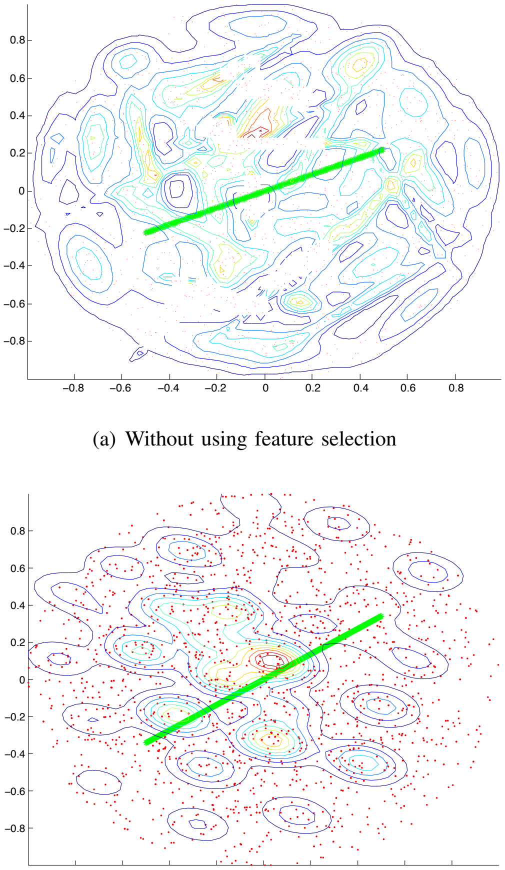
Before going on, let us observe that the final grade (Eqs. 6, 10) can be seen in terms of a distance from a hyperplane,

$$\begin{aligned} u(\bar{\boldsymbol{g}}, \boldsymbol{a}) &= \langle \boldsymbol{W}\bar{\boldsymbol{g}}, \boldsymbol{V}\boldsymbol{a} \rangle = \|\boldsymbol{W}\bar{\boldsymbol{g}}\| \|\boldsymbol{V}\boldsymbol{a}\| \cos(\boldsymbol{W}\bar{\boldsymbol{g}}, \boldsymbol{V}\boldsymbol{a}) \\ &= \|\boldsymbol{W}\bar{\boldsymbol{g}}\| \, \mathrm{d}(\mathrm{hyper}(\boldsymbol{W}\bar{\boldsymbol{g}}), \boldsymbol{V}\boldsymbol{a}). \quad (11) \end{aligned}$$

Thus, the representation of the words in the corpus in $\mathbb{R}^k$ is such that their distance to the hyperplane in $\mathbb{R}^k$ perpendicular to $\boldsymbol{W}\bar{\boldsymbol{g}}$ is proportional to its grade. In Figure 1, where $K = 2$, the green line is the hyperplane perpendicular to $\boldsymbol{W}\bar{\boldsymbol{g}}$. The points represented in the figure are those whose distance to the hyperplane is relatively large. There are several types of feature selection methods [14], namely filters, wrappers and embedded. Wrapper methods are general-purpose algorithms that search the space of feature subsets, testing performance of each subset using a learning algorithm. Embedded methods select features using an implicit learning process, and finally filters use general statistical tests to measure the relevance of feature/feature subsets. While wrappers (and embedded in a lesser extent), tend to overfit and are computationally intensive (although usually obtain good results), filters are independent of any learning method and less prone to overfit. For these reasons, we have concentrated on filters, and to this end, as there is no single dominant algorithm, we have chosen three popular feature selection methods based on Information Theory [15], named: Mutual Information Maximization (MIM), Joint Mutual Information (JMI), and minimum Redundancy Maximum Relevance (mRMR), for they obtain the best overall trade-off for accuracy/stability (JMI and MRMR) criteria, as stated by [15]. Finally, notice that all three methods provide an ordered ranking of *all* the features, so it is necessary to establish a *threshold* in order to select a reduced set of features. In this case, we have opted for retaining the top-n words with different values for n. Somehow, these words are the most representative.

The words selected in this way were used to initialize the peaks of a Gaussian Mixture Model (GMM) to fit the distribution of words. It is well-known that the selection of peaks to initialize a GMM is very important. In this paper we show that this selection can be accomplished by a Machine Learning tool, feature selection. For the sake of comparison, we also include the results when those peaks were chosen randomly; that is, when no feature selection was applied.

In Figure 2 we depict graphically how feature selection contributes to obtain an adequate solution. While the upper part of the figure shows an almost random fitness of Gaussians due to the fact that the initialization was really random, the lower part presents a reasonable distribution obtained with the assistance of a feature selection method. To be able to assess the goodness of the fitness, we will use standard measures detailed in the next paragraphs.



(a) Without using feature selection

(b) Using Feature Selection

Fig. 2. Scatter Plot and Fitted Gaussian Mixture Contours obtained for one of the data sets employed in the Experimental section: the AI exam (a) without using feature selection, (b) with the methodology described, removing stopwords, using the JMI method, and 50 top words as threshold

The complete methodology that was followed in this work, trying to automatically provide some feedback to the instructors, is detailed below:

1) Remove stop-words and words appearing only once in the corpus.
2) Learn matrices $\boldsymbol{W}$ and $\boldsymbol{V}$.
3) Sort list of words according to their grade (Eq. 10) and then select those in the first (Q1) and fourth (Q4) quartiles.
4) For each quartile Q1 and Q4,
   - Apply the FS methods (MIM, JMI, mRMR and no FS) to sort again words according to their *relevance*.
   - For each FS method and number of features (depending on the threshold used):
     - Fit a GMM distribution to data initializing the peaks of the Gaussians according to the top selected words.
     - Compute BIC (*Bayesian Information Criterion*) an AIC (*Akaike Information Criterion*) for each GMM [16].

- Choose the model with the lowest BIC and AIC values. For this model:
  - Sort the obtained clusters according to their probability density function (PDF) values.
  - Provide the instructor with the words conforming the first *3* clusters. Of course, this number can be modified, we chose 3 for short.

IV. Experimental Results

The general procedure detailed above has been tested over in pilots of peer-assessment in three different undergraduate courses: *Artificial Intelligence*, *Applied Economics* and *Constitutional Law*.

A. Datasets

The first dataset records the answers and grades obtained from an assignment of a course about *Artificial Intelligence* (AI). The assignment was common for three universities of Spain: University of A Coruña, University of Oviedo at Gijón and University Pablo de Olavide at Sevilla. In this experiment, the students of the 3 universities answered some questions in the field of blind and heuristic search methods. Specifically, the students were asked to use a tool called *AISpace* [17], and they should find the shortest path in a graph that represented Vancouver's neighborhoods (see Figure 3). The algorithms that should be employed to obtain such a path were already implemented in *AISpace*, and results should be justified by the students using optimality criteria of the algorithms.

All answers (175 in total) had been anonymized previously to send them to an Easy Chair event. After submission, the students took the role of graders (160 students), and were supplied with several answers (an average of 8.29) of other students (randomly and avoiding self-assessment), and with a rubric, containing detailed instructions on how to do the assessment of the assignments in a numeric scale of integers in the [0,10] interval. Each answer received 7.58 grades on average, and thus 1326 evaluations were obtained. The assessment matrix sparsity is high, as only 4.74% out of 28000 possible assessments were obtained. The characteristics of the dataset are shown in table I. This dataset was already used in [3], but the content of the answers was not addressed, nor its use for giving feedback to instructors.

The second and third datasets contain results of assignments carried out at the University of Oviedo (Uniovi) in a course of *Applied Economics* (AE) and *Constitutional Law* (C). In the first case, students should comment on an article [18] on the economic crisis in Spain, while in the second, available options for Parliamentary motions at the Spanish Parliament were to be discussed. The characteristics of these datasets (as in the case of the AI assignment), are also shown in Table I. As it can be seen, AI is the dataset with the highest number of total evaluations, followed near by AE and finally, C is the smallest dataset, with approximately half size than AI. However, the number of words used in each assignment follows the inverse order, being higher in C exam, that doubles AE, that in turn doubles AI. In both cases we used a Moodle[1] platform.

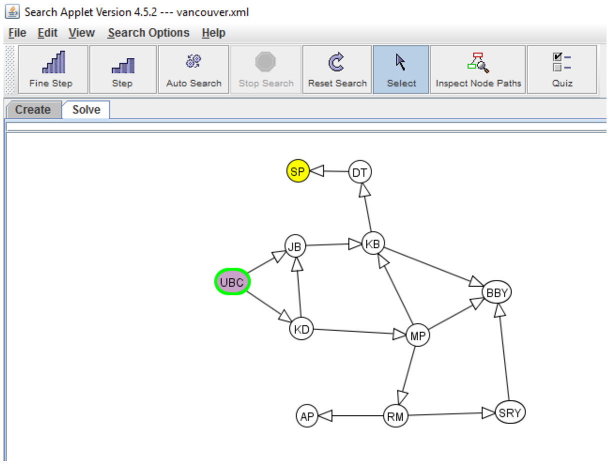


Fig. 3. Network of nodes used in the assignment for Vancouver's neighborhoods

TABLE I
A summary of the characteristics of the three datasets obtained from assignments of *Artificial Intelligence* (AI), *Applied Economics* (AE) and *Constitutional Law* (C), used in the experimental study

	AI	AE	C
Answers	175	111	66
Graders	160	108	66
Total evaluations	1326	1065	660
Sparseness (in %)	95.26	91.36	84.85
Avg. eval./ answer	7.58 ± 2.02	9.59±0.67	10.00±0.00
Avg. eval./grader	8.29 ± 1.45	9.86±0.99	10.00±0.00
Words	296	611	1112

B. Results

As stated in Section III, all feature selection filters used are rankers; that is, they return the complete set of features ordered. Thus, a threshold should be used in order to select a subset of features, in our case words.

Table II shows the best AIC and BIC values (of all thresholds) for each combination of feature selection method and quartile. Notice that we are including a row with the results achieved when no feature selection (no FS) was used to initialize the peaks of the Gaussians. Both AIC and BIC measures are desired to be minimized, and in these tables we can see that the minimum values are obtained by far when using feature selection methods.

The thresholds used in the experiments retained the top 15, 25, 35 and 50 words in the case of the assignment of *Artificial Intelligence*; and the top 15, 25, 50, and 75 in the case of *Applied Economics* and *Constitutional Law*. When no feature selection was applied, we randomly choose the same

[1] https://moodle.org

TABLE II
AIC AND BIC VALUES FOR *Artificial Intelligence* (AI), *Applied Economics* (AE) AND *Constitutional law* (C) ASSIGNMENTS, FOR THE THREE FEATURE SELECTION METHODS (MIM, JMI AND mRMR) AND WHEN NO FEATURE SELECTION (NO FS) IS USED. BEST RESULTS (LOWER) ARE MARKED IN BOLD FACE

		BIC		AIC	
Course	Filter	Q1	Q4	Q1	Q4
AI	No FS	-4250.6	-2771.7	-5085.9	-3330.1
	MIM	-5651.2	**-3978.7**	-6486.6	**-4771.8**
	mRMR	-5986.4	-3692.5	-6821.8	-4485.6
	JMI	**-6245.4**	-3688.0	**-7080.8**	-4481.2
AE	No FS	-12017.5	-9567.4	-13491.0	-10565.5
	MIM	**-16569.9**	**-15632.3**	**-18043.4**	**-17123.0**
	mRMR	-15133.6	-13848.4	-16607.1	-15339.1
	JMI	-13896.4	-13089.1	-15369.8	-14579.8
C	No FS	-10816.8	-8580.2	-12390.0	-10102.0
	MIM	-8684.1	-8770.5	-10176.2	-10292.3
	mRMR	**-13097.1**	**-13777.2**	**-14670.3**	**-15299.0**
	JMI	-8684.3	-8886.7	-10044.5	-10408.5

number of features than those retained by the feature selection methods.

Finally, let us review the words selected by our method in each of the datasets. In the case of the assignment of *Artificial Intelligence*, the students were asked to use three algorithms (namely, A^* with two different heuristics, and Breadth First), to solve the search problem mentioned in the beginning of this section, and then justify the results obtained. The discussion has to deal with the optimality of the paths and the complexity of the algorithms used. In Table III we record the words selected for feedback. All words relate to the specific names of the algorithms or to central concepts involved in search algorithms. The words included in the first quartile are most precise, specifically the first cluster contains the word that refers to the optimal algorithm (A^*), while the words included in the last quartile are more general. The term *BBY* is the acronym for a neighborhood that has some peculiarities in the performance of the searching algorithms; see Figure 3.

TABLE III
Artificial Intelligence ASSIGNMENT. FOR Q1, THE BEST RESULTS WERE OBTAINED FOR JMI FEATURE RANKER, USING THE TOP 50 WORDS AS THRESHOLD. FOR Q4 THE BEST RESULTS WERE OBTAINED BY MIM FEATURE SELECTION METHOD, USING THE TOP 30 WORDS.

Quartile	cluster	words
Q1	1	A*
	2	breadth
	3	nodes, weight
Q4	1	algorithm, prototype, solution
	2	BBY, destination, none, observe
	3	expanded, level

In the case of *Applied Economics* dataset (AE), students had to comment on an essay [18] on the economic crisis in Spain. The words that deserve the highest (Q1) and lowest (Q4) scores are shown in Table IV. In addition to temporary references to key periods to explain the economic situation, we find words like *crisis*, or *policy*, and again more specific terms in Q1 and more general terms in Q4.

TABLE IV
Applied Economics ASSIGNMENT. BEST RESULTS WERE OBTAINED FOR BOTH Q1 AND Q4 WITH MIM FEATURE SELECTION METHOD, USING THE TOP 75 WORDS.

Quartile	cluster	words
Q1	1	decade, states, moment, politics
	2	crisis
	3	XIX
Q4	1	policies
	2	growth
	3	fourth, external, measure

TABLE V
Constitutional Law ASSIGNMENT, THE BEST RESULTS WERE OBTAINED USING mRMR WITH 75 FIRST WORDS AS THRESHOLD FOR BOTH Q1 AND Q4.

Quartile	cluster	words
Q1	1	account, signature, mention, policy, reply situations, triumph
	2	111, affects, candidature, amend, moral, depart position, reasons, temporal, total
	3	10, 113, 35, author, concluded, necessarily, names objects, raise, proposer, practice, register
Q4	1	correspond, forty, eight
	2	101, accumulate, affirm, equipped, effect, f g, include, initiative, j, level
	3	lessened, separated, debate, dependence endowed, free-of-charge, represent

The assignment on *Constitutional Law* is to discuss the options of a government in Spanish Parliament on the investiture vote of confidence and dissolution of the Chamber. Students should argue on the legal requirements of these options; this meant in several cases to expressly cite articles of the Spanish Constitution or the Rules of Procedure of the Spanish Parliament. Table V shows the words used by answers in Q1 and Q4. We observe the words of the legal field together with the mention of key articles of the Constitutional Law (111, 10, 113, etc.), in the discussion that the students had to present.

V. CONCLUSIONS AND FUTURE WORK

Assessing automatically larger number of assignments, as it is necessary nowadays in MOOCs scenarios (among others) with thousands of students, has been dealt commonly by using multiple-choice tests. Nevertheless, there are some students' abilities and knowledge types that will be better judged using open-response questions. In this case, peer assessment has been the adopted strategy, as teachers can not possibly confront the situation. Thus, students are asked to assess a small portion of answers from their fellow peers, and using later a machine learning approach, learn a function that can palliate subjective scoring and other undesirable effects, and finally provide all students with a grade that give them feedback on their learning achievements. However, besides of students there are other actors in this play, the instructors, which feedback has been largely ignored by the ongoing research. The work described in this paper contains a proposal that tries to address this problem. In a previous work [3] we have ranked the students according to the grades that they obtained. Now, our rationale consists on analyzing also the use of the words in the students' answers, regarding the qualifications obtained, using a representation of words in a euclidean space that has semantic implications. Thus, the methodology proposed starts by sorting the list of words used according to the grades obtained by the students, in order to separate those used by the best and worst qualified assignments, first and fourth quartiles respectively. This step is followed by the application of a feature selection ranker for both Q1 and Q4 to sort again the words employed. This step is critical for an adequate initialization of the peaks of a GMM, that finally obtains a number of clusters (three, in our case, although this is configurable in the method) containing the most used words in the respective assignments. The instructors are thus provided with the words of those clusters in the respective quartiles (best and worst), in an attempt to supply a representation of the concepts learned by the students, and hopefully allowing them to possibly reorient materials and explanations for a more personalized learning for the students. To the best knowledge of the authors, this is the first attempt to provide feedback to instructors in the scientific literature.

To test the proposed methodology, three different datasets have been employed, in which students completed tasks in different disciplines, namely *Artificial Intelligence, Applied Economics* and *Constitutional Law*, each containing different numbers of words and assignments. The results obtained encourage us to pave the way for further developments in this endeavor. Accuracy and scalability of the method could be enhanced considering other possibilities for feature terms beside just individual words, one of them could be the inclusion of any consecutive sequence of word items (*n-grams*), which will be our next objective. Also, as future work, we plan to compare our work with other automatic methodologies to extract significant words from written texts.

Acknowledgment

This research has been financially supported in part by Spanish Ministerio de Economía y Competitividad (research projects TIN 2015-65069-C2, both1-R and 2-R), by European Union FEDER funds and by Xunta de Galicia (research project GRC2014/035). V. Bolón-Canedo acknowledges Xunta de Galicia postdoctoral funding (grant ED481B 2014/164-0). Additionally, we would like to acknowledge the Professors collaborating with us in this research: Alicia Troncoso (U. Pablo de Olavide), Juan Vázquez (U. Oviedo), and Francisco Bastida (U. Oviedo).

References

[1] J. Frederiksen and A. Collins, "A systems approach to educational testing," *Educational Researcher*, vol. 18, pp. 27–32, 1989.

[2] S. Gielen, F. Dochy, P. Onghena, K. Struyven, and S. Smeets, "Goals of peer assessment and their associated quality concepts. studies in higher education," *Studies in Higher Education*, vol. 36, pp. 719–735, 2011.

[3] O. Luaces, J. Díez, A. Alonso-Betanzos, A. Troncoso, and A. Bahamonde, "A factorization approach to evaluate open-response assignments in moocs using preference learning on peer assessments," *Knowledge-Based Systems*, vol. 85, pp. 322–328, 2015.

[4] K. Raman and T. Joachims, "Methods for ordinal peer grading," in *ACM Conference on Knowledge Discovery and Data Mining, KDD*, 2014.

[5] N. B. Shah, J. K. Bradley, A. Parekh, M. Wainwright, and K. Ramchandran, "A case for ordinal peer-evaluation in moocs," in *NIPS Workshop on Data Driven Education*, 2013.

[6] P. F., N. Tishby, and L. Lee, "Distributional clustering of English words," in *Proceedings of the ACL*, 1993, pp. 183–190.

[7] L. Baker and A. K. McCallum, "Distributional clustering of words for text classification," in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 1998, pp. 96–103.

[8] N. Slonim, N. Friedman, and N. Tishby, "Unsupervised document classification using sequential information maximization," in *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 2002, pp. 129–136.

[9] G. McLachlan and D. Peel, *Finite mixture models.* John Wiley & Sons, 2004.

[10] G. Salton, A. Wong, and C. S. Yang, "A vector space model for automatic indexing," *Commun. ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[11] P. Turney and P. Pantel, "From frequency to meaning: Vector space models of semantics," *Journal of Artificial Intelligence Research*, vol. 37, pp. 141–188, 2010.

[12] D. Forman, *Feature Selection for Text Classification*, ser. Computational Methods of Feature Selection. CRC Press. Chapman and Hall, 2007, pp. 2–21.

[13] Y. Yang and J. Pedersen, "A comparative study on feature selection in text categorization," in *International Conference on Machine Learning*, 1997, pp. 412–420.

[14] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *The Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[15] G. Brown, A. Pocock, M. Zhao, and M. Luján, "Conditional likelihood maximisation: a unifying framework for information theoretic feature selection," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 27–66, 2012.

[16] G. Box, G. M. Jenkins, and G. Reinsel, "Time series analysis: Forecasting & control," 1994.

[17] B. Knoll, G. Kisynski, G. Carenini, C. Conati, A. Mackworth, and D. Poole, "Aispace: Interactive tools for learning artificial intelligence," in *Proceedings of the AAAI 2008 AI Education Workshop*, 2008.

[18] J. A. V. García, "Las décadas ganadas y perdidas de la economía española," in *Ensayos sobre economía española: homenaje a José Luis García Delgado.* Civitas, 2014, pp. 25–34.