Modeling peer assessment as a personalized predictor of teacher's grades

The case of OpenAnswer

Maria De Marsico, Andrea Sterbini
Computer Science Dept.
Sapienza University
Rome, Italy
{demarsico, sterbini}@di.uniroma1.it

Marco Temperini
Dept. of Computer, Control and Management Engineering
Sapienza University
Rome, Italy
marte@dis.uniroma1.it

***Abstract*— Questions with open answers are rarely used as e-learning assessment tools because of the resulting high workload for the teacher/tutor that should grade them. This can be mitigated by having students grade each other's answers, but the uncertainty on the quality of the resulting grades could be high.**

In our OpenAnswer system we have modeled peer-assessment as a Bayesian network connecting a set of sub-networks (each representing a participating student) to the corresponding answers of her graded peers. The model has shown good ability to predict (without further info from the teacher) the exact teacher mark and a very good ability to predict it within 1 mark from the right one (ground truth). From the available datasets we noticed that different teachers sometimes disagree in their assessment of the same answer. For this reason in this paper we explore how the model can be tailored to the specific teacher to improve its prediction ability. To this aim, we parametrically define the CPTs (Conditional Probability Tables) describing the probabilistic dependence of a Bayesian variable from others in the modeled network, and we optimize the parameters generating the CPTs to obtain the smallest average difference between the predicted grades and the teacher's marks (ground truth). The optimization is carried out separately with respect to each teacher available in our datasets, or respect to the whole datasets.

The paper discusses the results and shows that the prediction performance of our model, when optimized separately for each teacher, improves against the case in which our model is globally optimized respect to the whole dataset, which in turn improves against the predictions of the raw peer-assessment. The improved prediction would allow us to use OpenAnswer, without teacher intervention, as a class monitoring and diagnostic tool.

Keywords—Modeling peer-assessment; Bayesian networks; Automatic correction of open answers;

I. Introduction and Related Work

Peer assessment is seldom used and hard to correct. It is considered a useful exercise to challenge as well as improve one's understanding of a topic but also to achieve higher metacognitive abilities. Actually, according to Bloom's taxonomy of educational objectives in the cognitive domain [3], learner's abilities increase when passing from pure knowledge (the ability to remember a topic is at the lowest level), to comprehension, application, analysis, evaluation and finally synthesis. In [1] a revised version of the taxonomy is proposed, where remember, understand and apply lay at increasing levels, while analyze, evaluate and create lay at the same top level. In any case, the ability to evaluate is a higher metacognitive skill going beyond the proficiency in a single topic, though requiring it. As a matter of fact, as discussed in [13], metacognitive activities require not only knowing but also knowing about knowing. The accepted definition of metacognition refers to higher order thinking, entailing the ability to exercise an active control over the cognitive processes underlying learning. Planning strategies and schedules to carry out a learning task, monitoring one's and others' comprehension of a topic and the progress towards the completion of a task, and being aware of how to apply newly acquired concepts and rules, all play a critical role in successful learning. Peer assessment can be exploited to this aim.

The OpenAnswer (OA for short) framework [17][18][19] used in the experiments presented in this paper allows (semi-)automated grading of open answers through peer assessment and to model/analyze the class knowledge level, with the further goal of relieving the teacher from part of the grading burden. As a matter of fact, while this kind of exercise provides a much more reliable evaluation of students' proficiency with respect to, e.g., multiple-choice tests [15], they are also much more demanding for the teacher too, since they require a long correction activity.

During an OA assessment session, each student is requested to grade some (e.g., 3) of her peers' answers. The validity of results of peer evaluation could be enforced by requiring that a subset of answers (chosen according to some relevant strategy) is further graded by the teacher.

In this paper we choose to investigate how much the model is able to predict the correct grades without any teacher intervention, right before the correction phase. Our main goal being to apply OA as a peer assessment monitoring and analysis tool which could be used without teacher intervention on communities of learners.

Assessments provided by peers (and the teacher if present) are fed and propagated within a Bayesian Network (BN) made of interconnected discrete variables. In such network the students are modeled by their Knowledge level on the topic (K), and by their ability to do their evaluations, denoted as Judgment (J). In the network, the answers of a single student have an estimated Correctness (C), which can be updated by evidence

propagation. When a student marks a peer's answer, a corresponding Grade (G) is injected into the network, and propagates its effects depending on both J of the grading student and on current estimation of C of the answer corrected. Variables C and J are assumed to be conditioned by K (C | K and J | K), and G by J and C (G | J, C), therefore for each of them we have a Conditional Probability Table (CPT) describing the corresponding probabilistic dependence from the values of the parent variables.

If the resulting analysis is shown in class, students can both better understand how the grading process should work, by matching the grades they assigned with final ones (possibly by the teacher, or inferred by the system through the BN), and learn from smarter peers how to improve their results [16]. Providing the students with their final K and J values, besides the pure exercise grade C, could spur further metacognitive awareness.

A. *Related work*

Automatic analysis of open answers is a powerful means to manage assessment in education, also known as knowledge tracing [2]. In other fields, such as in a context of marketing applications, where techniques of data mining and natural language processing are used to extract customer opinions and synthesize products reputation [22]. In [11] concept mapping and coding schemes are used with the same goal. (Semi-)automatic assessment of open-answers proposed in [4] relies on ontologies and semantic web technologies. Ontology models the knowledge domain related to the questions, and also aspects of the overall educational process. In [9] open answers are examined to identify and treat students misconceptions which hinder learning.

Peer-assessment is the activity in which a learner, or a group of learners, assesses the product of other learners (the peers) which is a higher cognitive level activity [3]. Peer-assessment can be used to pursue both formative and summative goals [20]: in the first case the aim is to allow the learner to appreciate her cognitive situation (such as level of knowledge, or lacks therein) and monitor her progress. In the second case not all the available information might reach the learner, and the aim is to evaluation and possible support to the selection of remedial activities. Li et al. in [12] states that a relationships does exist between the quality of the peers feedback, on a learner's job, and the quality of the final project submitted by the learner. A comprehensive study of peer assessment in a prototype educational application is in [6].

Our OA system relies on the evaluation of answers coming from peer-assessment, and on student modeling managed by Bayesian Networks. Another machine learning approach to student modeling is in [7], where Bayesian Network techniques are used to support learner's modeling in an Intelligent Tutoring System (ITS). There, modeling is devised to support activities relevant in an ITS: knowledge assessment, plan recognition and prediction, the last two deemed to see what intentions are behind a learner's choice, and what following choices might be, during the phase of problem solving. In OA the peer is presented with a set of assessing criteria, to refer to while marking; the criteria are defined by the teacher. In our experience too many criteria might result cumbersome for the peers. We have not investigated, though, on this aspect. In literature the specificity of "scoring criteria" has been identified as an important factor against the problem of having assessors that limit the range of their marks to a subset (typically in the high end) of the scale; in this case the problem is twofold, involving both peers leniency and shrinking of the marking scale [14]. An aspect of research in peer-assessment regards the number of peer-evaluations that a same job should undergo during the peer-evaluation process. In OA this is configurable, with a default of 3. In literature it is found that more feedback on the same job make the peer performing more complex revisions on her product, and ending up with a better result [5].

II. The OpenAnswer model

The OA system models peer-assessment as a Bayesian network made of interconnected sub-networks, each one representing one of the participating students. The student model sub-network is made of three discrete nodes/variables, representing respectively:

- ***K***: her **knowledge** about the topic
- ***C***: the **correctness** of her answer
- ***J***: her ability to **judge**/assess the answer of a peer
- plus one variable ***G*** for each **grade** given to a peer

Each Bayesian variable above has 6-valued discrete domain ranging from A (best) to F (fail).

We assume that both *C* and *J* probabilistically depend on *K* because 1) *C*: writing an essay cannot easily guessed as in multiple-choice quizzes (we do not model cheating yet); 2) J: we are inspired by Bloom's taxonomy of cognitive levels [3] assuming that judging a peer's answer should be a more difficult task than knowing the topic and answering it. To complete the student sub-network, we assume that the Grade given to a peer's answer probabilistically depends both from its Correctness and from the student's Judgment ability.

Fig. 1 shows an example of a student sub-network, with the probability computed for each domain value.

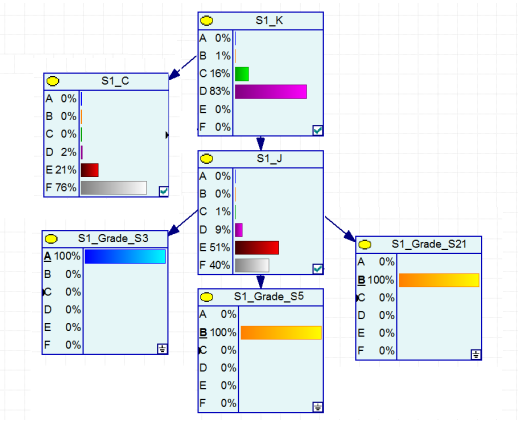

Fig. 1: Bayesian model of student S1, grading the answers of students S3, S25 and S21 with grades A, B, B, respectively.

Once the Bayesian network is complete we use it to infer what would be the K, J and C of each student, by adding as

evidence all the G peer grades and by propagating the values in the Bayesian network. Our aim is to analyze the assessment and possibly get a clear picture of the class modeled. Moreover, with a partial correction from the teacher, we could refine the network prediction of the student's C by entering as evidence the actual Correctness of the subset of corrected answers, i.e. the teacher's grade. In earlier works [8][17][18][19] we analyzed several parameters affecting the network by simulating the teacher's correction, and we have noticed that the CPTs (Conditional Probability Tables), which define the probabilistic dependencies of each variable from her parents (J and C from K, and G from J and C) influences the quality of the predicted grades.

This brought us to the investigation which is the topic of this paper, where we try to learn the CPTs from our datasets to possibly get the best predictions. In doing this we noticed also that the model shows different precision depending on the teacher. In this paper we analyze the raw prediction performances (without any teacher correction), and compare them both with:

- the average peer grades for each answer (without the OA model),
- the overall performance of OA when optimized over the whole available datasets, i.e., when we try to find a model that fits all teachers at the same time.

Thus, our research questions are:

RQ1: how much the OA model improves its prediction respect to the ability to predict the teacher grades by the peers, when the parametric CPTs used in the network are optimized over the whole dataset?

RQ2: then, does the OA model further improves its predictions, when the parametric CPTs used in the network are optimized separately for each teacher present in the dataset instead than globally respect to the whole dataset?

III. METHODOLOGY

Learning Bayesian network's CPTs can be done by appropriate machine learning (ML) algorithms. This has the problem that it requires a great amount of example data from which to learn. Moreover, the whole OA peer-assessment network is made of too many nodes. Even if, instead than learning the whole network, we restrict ourselves to learning just a general model of how the student behaves in a peer-assessment, i.e. just the student sub-network, we would still have too many parameters to learn (30 values for each J and C, 180 for G) for the data available. Moreover, the data available is both not big enough and, most important, does not cover all the possible combinations of variable values, making very difficult to learn the complete CPTs by ML.

These limits has pushed us to try to lower the number of parameters to be learned, by imposing some kind of structure over the CPTs. To this aim we have chosen to define the probability distribution of a depending Bayesian variable Y (for each possible value X of the parent variable), that is a column in the CPT, as a Gaussian normal distribution with *mu, sigma* parameters tied to the parent variable value X as follows:

$$P(J \mid K) = Gauss(J, sigma(K), mu(K))$$

$$P(C \mid K) = Gauss(C, sigma(K), mu(K))$$

$$P(G \mid J, C) = Gauss(G, sigma(J, C), mu(J, C))$$

where we define *sigma* and *mu* as the following linear functions of K (and J, C):

$$sigma(K) = a*K + b$$

$$mu(K) = c*K + d$$

$$sigma(J, C) = e*J + f*C + g$$

$$mu(J, C) = h*J + i*C + j$$

The rationale being that in this way we are building CPTs where both the depending variable **value** *mu* and its **error** *sigma* are linearly dependent on the parent values. In facts, for any given K we model the depending variable (J or C) as a value *mu(K)* within a *sigma(K)* error as defined above, both linearly dependent on K (and similarly for the definition of P(G|J,C) respect to J and C). This allows us to reduce the number of parameters required to define the J and C CPTs from 30 to 4 each and the parameters defining the G CPT from 180 to 6. An example CPT for P(J|K) generated by the parameters [a=0.70 b=0.50 c=0.21 d=0.32] is show in Tab. I, where the maximum of each distribution for each K value (column) is highlighted (bold).

TABLE I. EXAMPLE P(J|K) GENERATED BY PARAMETERS a=0.70 b=0.50 c=0.21 d=0.32

P(C | K)	K					
	A	B	C	D	E	F
A	**60.9%**	16.5%	0.2%	0.0%	0.0%	0.0%
B	22.4%	**36.5%**	14.1%	1.0%	0.0%	0.0%
C	8.8%	23.5%	**30.4%**	14.8%	2.9%	0.2%
D	4.2%	12.6%	26.3%	29.9%	19.4%	7.5%
E	2.3%	6.9%	17.7%	**30.4%**	36.8%	33.0%
F	1.4%	4.0%	11.2%	23.9%	**40.9%**	**59.2%**

Given the 12 coefficients *a,b,…,j*, we generate the CPTs of the student's sub-network and then the whole network, which is used (together with the G peer grades as evidence) to predict the C grades. These are compared to the Teacher's grades (ground truth) and the average absolute difference *AvgDeltaCV* is computed for the whole assessment.

$$AvgDeltaCV = Avg(\mid PredictedGrade - TeacherGrade \mid)$$

To find the best 12 *a,b,…,j* parameters we optimize by using the Adaptive Simulated Annealing optimization library [10] to minimize *AvgDeltaCV* over the set of peer-assessments made by the same teacher. The optimization runs for max 6 hours and then is stopped, providing the current best solution.

The available datasets comes from different areas, with questions both at high-school and at university level (see Tab. II). The groups of students range from a minimum of 5 to a maximum of 60. For computational reasons we split the biggest groups into highly connected sub-networks of maximum 12

students (this was the case for the question in dataset A2 and for two of the questions in dataset A).

TABLE II. THE COMPOSITION OF THE USED BENCHMARK DATA.

Dataset	Level	Topic	Groups	Students
A	Univ.	12 exercises on multi-level cache systems	1	5 to 15
M	Univ.	3 exercises on C programming	2	9 to 13
I	High School	1 physics exercise	2	14 and 12
A2	Univ.	1 essay on social tools	5	60 split in 5 groups of 12
F	High School	1 q. on numbers' representation	2	10-12

By comparing the student's grades with the teacher's grade we can measure the error of the average peer grades *AvgDeltaPeerGrade* depending on the teacher, which is shown in Tab. III.

$$AvgDeltaPeerGrade = Avg(\,|Avg_i\left(PeerGrade_{i_j}\right) - TeacherGrade_i\,|)$$

where

$TeacherGrade_i$ is the grade given to student i

$PeerGrade_{i_j}$ is the grade peer j gave to student i

TABLE III. AVERAGE PEER ERROR FOR EACH DATASET/TEACHER

DATASET	TEACHER	AvgDeltaPeerGrade
A	**A**	1,22
A2	**5**	1,38
	887	1,73
	1033	1,62
F	**F**	1,32
I	**I**	0,50
M	**M**	1,20

Except for the case of dataset I, the students are definitely more than one grade off respect to the correct grade. In particular, teacher 887 in dataset A2, is the one most disagreeing with the student's assessment (1.73 difference), closely followed by teacher 1033 (1.62).

IV. RESULTS

The resulting optimized errors *AvgDeltaCV* are shown in Tab. IV, where the best/lowest values are highlighted (bold).

In particular, in our experiments we have tried two different initializations for the probability distribution P(K) of the independent variable K with the (added) goal to see which initialization behaved better:

- **flat:** constant probability = 1/6 (this to model when the system has no knowledge about the class)
- **TgradeDist:** the same probability distribution as we get from the teacher grades of that assessment (this to show what would happen if the system had some initial global information on the class but no personal information on each student)

TABLE IV. AVG. PREDICTION ERRORS FOR EACH DATASET/TEACHER VS P(K) INITIALIZATION

AvgDeltaCV		STAT		
DATASET	**TEACHER**	**flat**	**TgradeDist**	**Variation**
all	**any**	1.10	1.01	7.7%
A	**A**	0.96	0.94	1.4%
A2	**5**	1.09	1.05	4.0%
	887	1.38	1.34	3.0%
	1033	0.94	1.03	-9.3%
F	**F**	0.70	0.70	-0.7%
I	**I**	**0.36**	**0.35**	3.1%
M	**M**	1.08	0.89	**17.8%**

In Tab. IV we show the relative improvement of the prediction error (*Variation* column) when we move from *flat* to *TgradeDist* initialization. As one could expect, the *TgradeDist* initial P(K) allows the network to predict with smaller errors, decreasing them in the best case by 17% (except for teacher 1033, which could be the result of the optimization getting stuck in a local minimum).

The resulting optimized CPTs allows us to obtain an average error in some cases (teacher I) as low as 0.3 marks, in general up to 1 grade off and only in one case significantly more than 1 grade off (teacher 887). To explain this particular case, notice that the A2 dataset (see [21]) is based on one single peer-assessment which has been graded by three different teachers rather differently from what the peers did (and from each other), as we have seen in Tab. III. When optimized over the whole group of datasets (line "all-any") the network predicts grades in average 1 grade off from the ground truth.

When we compare the obtained OA average error with the error given by the peer-assessment only (as shown in Tab. V, where the best values are highlighted in bold), we get a huge improvement, as we reduce the error by at least 10%, and up to 47% in the best case of teacher F with flat P(K). In the general case (line all-any), where the optimization is done over the whole set of teachers and questions, the error is at least 17% lower for flat P(K), and 24% better for TgradeDist P(K).

TABLE V. OPTIMIZED ERROR VS PEER ERROR.

DATASET	TEACHER	Average	Average DeltaCV		Improvement	
		DeltaPeerG	flat	TgradeDist	flat	TgradeDist
A	**A**	1,22	0,96	0,94	21,5%	22,6%
A2	**5**	1,38	1,09	1,05	20,8%	23,9%
	887	1,73	1,38	1,34	20,3%	22,7%
	1033	1,62	0,94	1,03	41,7%	36,2%
F	**F**	1,32	0,70	0,70	**47,0%**	**46,6%**
I	**I**	0,50	**0,36**	**0,35**	28,4%	30,6%
M	**M**	1,20	1,08	0,89	10,4%	26,3%
all	**any**	1,34	1,10	1,01	17,8%	24,1%

In particular, the very best performance of OA respect to teacher **I** could be also influenced by the fact that in that case the peer assessment was already very near the teacher grades (which in turn could perhaps depend on the teacher not having used the full grade range), more investigation on this case is required.

The main result is the improvement in prediction ability when the OA model is optimized for the teacher. We are confirmed in the fact that the lowest errors are obtained when the OA CPTs are separately optimized respect to the given teacher, and that the OA model greatly increases the precision of the predicted grades respect to the peer grades, even when no input from the teacher is used, as we are showing with these experiments.

V. Conclusions and Future work

In this paper we have shown that, even when the CPTs are generated by linear relations among the Bayesian variables from a small set of learned parameters, a good prediction ability can be observed, way better than the raw grades of the students, answering affirmatively to RQ1. Moreover an even better prediction is obtained when the OA model is specifically optimized for the teacher (RQ2).

We have initially chosen to base our CPTs on Gaussian distributions with parameters linearly depending from the parent variables, and to optimize the parameters by minimizing the average prediction error *AvgDeltaCV*. These choices imply that:

- For each value of K, the corresponding probability distribution is symmetric. This could be a too strong constraint, as one could expect that the probability of over-grading a peer would be different from that of under-grading her.
- The relation between K (and J, C) and the *mu* and *sigma* Gaussian parameters is linear, which also could be a too strict constraint on the model.
- By averaging the absolute errors, few big prediction mistakes (with higher *DeltaCV*) are treated the same than many smaller *DeltaCV*. This puts lower pressure on the highest errors. As, in our opinion, it is more important to reduce the biggest errors, it could be better to weight them more than the other in the optimized objective function.

For this reason in our following investigation we are improving our parametric definition of the CPTs to better fit with the data:

- by using a more general asymmetric Gaussian function (depending also on a *skew* parameter),
- by using higher grade polynomials to compute *mu*, *sigma* and *skew*,
- by optimizing the average square difference between predicted grades and ground truth to put "higher pressure" on reducing larger prediction mistakes.

Moreover, we have observed lower errors when the CPTs are optimized respect to the given teacher. This could depend on her ability to explain to students how to do the correction, or on the relative uniformity of tasks in the available dataset (which is made of similar exercises for each teacher). More investigation should be carried on, e.g. by adding to the answers already present in the datasets new corrections from other teachers or by adding new types of exercises.

Finally, we want to mimic a real usage of OA in an evolving setting, where the teacher adds new peer-assessments to her datasets and then re-optimizes the CPTs, to study the prediction's precision evolving in time.

References

[1] Anderson, L. W., Krathwohl, D. R. (eds.), 2000. A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives. Allyn and Bacon.

[2] Anderson, J.R., Corbett, A.T., Koedinger, K.R., Pelletier, R., 1995. Cognitive Tutors: Lessons Learned. The Journal of the Learning Sciences, 4(2), 167-207.

[3] Bloom, B.S., Engelhart, M.D., Furst,. E.J., Hill, W.H., Krathwohl, D.R., 1956. Taxonomy of educational objectives: The classification of educational goals. Handbook I. McKay.

[4] Castellanos-Nieves, D., Fernández-Breis, J., Valencia-García, R., Martínez-Béjar, R., Iniesta-Moreno, M., 2011. Semantic Web Technologies for supporting learning assessment, Inf. Sciences, 181:9.

[5] Cho, K., MacArthur, C., 2010. Student Revision with Peer and Expert Reviewing. Learning and Instruction 20(4).

[6] Chung, H., Graf, S., Robert Lai, K., Kinshuk, 2011. Enrichment of Peer Assessment with Agent Negotiation. IEEE TLT Learning Technologies, 4(1), pp.35-46.

[7] Conati, C., Gartner, A. , Vanlehn, K., 2002. Using Bayesian Networks to Manage Uncertainty in Student Modeling. User Modeling and User-Adapted Interaction 12, pages 371-417.

[8] De Marsico, M., Sterbini, A., Temperini, M., 2015. Towards a quantitative evaluation of the relationship between the domain knowledge and the ability to assess peer work. Proc. ITHET 2015 (pp. 1-6). IEEE.

[9] El-Kechaï, N., Delozanne, É., Prévit, D., Grugeon, B., Chenevotot, F., 2011. Evaluating the Performance of a Diagnosis System in School Algebra, ICWL, LNCS 7048.

[10] Ingber, L. "Adaptive Simulated Annealing (ASA),". Global optimization C-code, Caltech Alumni Association, Pasadena, CA, (1993).

[11] Jackson, K., Trochim, W., 2002. Concept mapping as an alternative approach for the analysis of open-ended survey responses. Organizational Research Methods, 5, Sage.

[12] Li, L.X., Liu, X. , Steckelberg, A. L., 2010. Assessor or Assessee: How Student Learning Improves by Giving and Receiving Peer Feedback. Br. J. of Ed. Tech. 41 (3), pages 525–536.

[13] Metcalfe, J., Shimamura, A. P., 1994. Metacognition: knowing about knowing. Cambridge, MA: MIT Press.

[14] Miller, P., 2003. The Effect of Scoring Criteria Specificity on Peer and Self-assessment. Assessment & Evaluation in Higher Education, 28/4.

[15] Palmer, K., Richardson, P., 2003. On-line assessment and free-response input-a pedagogic and technical model for squaring the circle. In Proc. 7th CAA Conf. (pp. 289-300).

[16] Sadler, P. M., E. Good, P. M., 2006. The Impact of Self- and Peer-Grading on Student Learning. Ed. Ass., 11(1).

[17] Sterbini, A., Temperini, M., 2012. Dealing with open-answer questions in a peer-assessment environment. Proc. ICWL 2012. LNCS, vol. 7558, pp. 240–248. Springer, Heidelberg.

[18] Sterbini, A., Temperini, M., 2013a. OpenAnswer, a framework to support teacher's management of open answers through peer assessment. Proc. 43th Frontiers in Education (FIE 2013).

[19] Sterbini, A., Temperini, M., 2013b. Analysis of OpenAnswers via mediated peer-assessment. Proc. 17th IEEE Int Conf. on System Theory, Control and Computing (ICSTCC 2013).

[20] Topping, K., 1998. Peer assessment between students in colleges and universities, Rev. of Ed. Research, 68, pp. 249–276.

[21] Vozniuk, A., Holzer, A., and Gillet, D. 2014. Peer assessment based on ratings in a social media course. In *Proceedings of the Fourth International Conference on Learning Analytics And Knowledge* (LAK '14). ACM, New York, NY, USA, 133-137.

[22] Yamanishi, K. , Li, H. , 2002. Mining Open Answers in Questionnaire Data, IEEE Int. Systems, Sept-Oct, pp 58-63.