Usage of Graph Patterns for Concept Map Extension

Maija Strautmane, *Riga Technical University, Latvia*

Abstract. **In concept map-based assessment an expert's concept map can be expanded using graph patterns to add hidden and inverse relations. This helps to avoid forcing a learner to use certain structures and names. Graph patterns are subgraphs that describe combinations of concept map elements, from which extra relations can be inferred. In this paper an enriched set of graph patterns is described along with their respective IF...THEN rules which can be used for automated knowledge assessment. Some of them are already implemented in the intelligent and adaptive knowledge assessment system IKAS.**

Keywords: **Concept map, graph pattern, hidden relationship, inverse relationship, concept map extension.**

I. Introduction

Advances in information and communication technologies change our everyday life, offering solutions for constantly growing range of application domains. Teaching, learning and knowledge assessment process is not an exception. Information and communication technologies enable student-centred, one-to-one and group learning in traditional and in computational environments [1]. In [2] Novak states that each educational event consists of five elements: a teacher, learner, knowledge, context and evaluation, which interact to create the meaning of experience. These elements can interact in virtual learning environments, as well as in traditional classroom settings.

In the learning process, regular knowledge assessment activities are necessary to timely diagnose potential misconceptions and correct them. This increases teachers' workload because the tasks for each knowledge assessment activity must be prepared beforehand and each learner's results must be evaluated. To reduce the teachers' workload, automatic knowledge assessment systems can be used and, consequently, a teacher has more time to help each individual learner to master the new knowledge. Thus, the aim of developing an automated scoring mechanism is to find a balance between as complete information about the learner's knowledge as possible and less workload concerning knowledge assessment for a teacher.

Evaluation of learner's knowledge in computer assisted assessment systems can be implemented by using rather wide range of questions. Depending on the type of questions used, these assessment systems can be classified as objective testing systems or subjective testing systems [3]. In the objective testing systems answers to questions are selected from or compared to a limited set of predefined responses. Examples of tasks used for this kind of systems are multiple-choice questions. Subjective testing systems are more difficult to implement because they have to assess the learner's response by content, style and originality. However, subjective testing systems do not restrict the learner's answers and, thus, allow assessing higher order skills [3]. Examples of questions in subjective testing systems are essays, free-text responses, as well as responses in a form of a concept map (CM).

Novak's research group introduced CMs in 1970s [2]. Since the beginning they have been used as a tool in education field. CMs are based on Ausubel's assimilation theory that posits a hierarchical memory structure and Deese's associationist memory theory that considers memory to be a set of interrelated concepts that can be arranged in a hierarchical as well as non-hierarchical structure [4].

CM is a graphical representation of a conceptual structure in the form of a graph. Nodes of the graph represent concepts, while arcs of the graph represent relationships between concepts. A graph can be directed or undirected and can have labels on links that describe the type of relationship between concepts. No concept has the meaning on its own; instead a concept meaning is defined by its relations to other concepts [4]. Propositions are basic units of meaning in CMs. They are statements about some object or event in the universe, either naturally occurring or constructed and can consist of two or more concepts [5]. Usually CM-based knowledge assessment systems analyze propositions that contain two concepts and the relation between them.

In computer-based CM scoring correctness of a proposition is usually determined by comparing it to one or several expert's CMs. If a learner gets scored only for those constructs that are exact matches of an expert's constructs, he/she is forced to use the same names and to connect concepts similarly to the expert instead of freely expressing his/her knowledge. This is in conflict with the cognitivist postulate that each human constructs his/her knowledge structure differently, according to his/her previous experience [6].

At the Department of System Theory and Design of Riga Technical University, a CM-based intelligent and adaptive knowledge assessment system IKAS has been developed since 2005 [7]. The author of this paper has been working in IKAS development group since 2007. IKAS uses graph patterns to expand an expert's CM before comparing it to a learner's CM. This approach allows adding correct relations that were not included in the expert's CM, but could appear in the learner's CM and should be scored. Additional relationships are derived basing on a combination of CM elements and without analyzing the semantics of concepts; thus, it does not require a natural language processing ability. Next section of this paper gives an overview of IKAS. Related works are overviewed in the third section. The fourth section explains the kinds of graph patterns that can be used to supplement scoring mechanism. Graph patterns for inference of hidden relationships are described in the fifth section, and graph patterns for inference of inverse relations are explained in the sixth section. The conclusions and some ideas for the future work are presented in the last section of this paper.

II. Intelligent knowledge asessment system IKAS

A detailed description of IKAS and its evolution is available in sources [7] and [8]. Here is a brief introduction to the working principles and main characteristics of the system.

To prepare the tasks for an assessment activity for a certain course, a teacher has to draw a CM that will be used as a base for the task generation and as an etalon for comparison with a learner's CM. The teacher can divide a course into several stages, where a stage can be any logically completed part of the course [7], such as a topic or a chapter. The teacher must create a CM for each stage in such a way that the CM of a particular stage is an extension of the CM from the previous stage. The learner completes the tasks sequentially stage by stage.

When the learner starts to solve a task of the next knowledge assessment or self-assessment stage, he/she can receive a task that corresponds to one of six available task difficulty degrees. Tasks vary from "fill-in-the-map" tasks that are easier (tasks of 1st to 4th degree) to "construct-a-map" tasks that are more difficult (tasks of 5th and 6th degree). A list of concepts that must be either inserted into the given structure or used to construct a CM is given for tasks of every degree of difficulty. A list of linking phrases is given only for tasks of 4th and 6th difficulty degree, where the learner has to label the relationships between concepts. For the 1st and 2nd degree tasks, linking phrases are already inserted into the given structure, but for the 3rd and 5th degree tasks no linking phrases are used at all. The system generates the task of appropriate degree of difficulty considering the teacher's predefined degree of difficulty and the learner's level of knowledge. For task generation the system uses the teacher's created CM. The learner can decrease the degree of task difficulty if he/she is not able to solve it, in this case the score will be increased when the learner has successfully finished the previous stage and the score exceeds a predefined threshold.

After the learner has submitted the solution of a task, the system compares it with the expert's CM that has been extended using graph patterns (currently only those are implemented that allow revealing hidden relationships, see Section 3). In IKAS those learner's created propositions that do not match any constructs in the expert's CM are delivered to the teacher. The teacher can then decide if this proposition is truly erroneous or it is valid for the given domain. Usage of graph patterns also increases the automation degree of CM scoring because there are less such propositions that cannot be evaluated automatically.

While solving a task, the learner can use several kinds of help: explanation of a concept, insertion of an additional concept into its correct place (if a CM structure is given), checking the correctness of a proposition and reduction of task difficulty. After submission of a solution, the learner also receives rich feedback that contains not only the gained score, but also the correctness of each proposition, concept mastering degrees for each concept, an individual study plan, type and amount of help used on solving a task and other information. Detailed information on IKAS provided feedback is presented in [9].

Currently there are six predefined types of linking phrases used in IKAS: "is-a" – denotes relationship between a class and its subclass, "kind-of" – relationship between concepts from two adjacent levels of hierarchy, "part-of" – relationship between the whole and its part, "example" – relationship between a class and its instance, "attribute"– relationship between an object and its attribute, "value" – relationship between an object attribute and its value. The teacher can use other linking phrases as well if he/she feels that none of those predefined relation types describe the relation accurately. These are called linguistic relations.

Examining the usage of various linking phrases in expert CMs drawn using IKAS, it has been found that about one third of all relationships are linguistic ones [10]. Frequency of other types of relations: "is-a"–18.9%, "kind-of" – 5.9%, "part-of" – 20.1%, "example" – 8.2%, "attribute" – 9.7%, "value" – 3.2%. Graph patterns describe only combinations of predefined linking phrases; thus, on average about one third of the expert's CM cannot be used for inference of extra relations. As it limits the degree of automation of the scoring mechanism, the survey has been performed to find other linking phrases that describe important conceptual relationships. As a result, 7 additional linking phrases are proposed for inclusion in IKAS as predefined relation types [10]. These are: "before", "located-at", "cause", "affect", "has-function", "made-of", "possesses". The set of linking phrases is still left open because there could be other types of relations that more precisely describe some particular connections between concepts.

III. Related work

There are not many attempts to automatically expand an expert's CM. However, in the C-TOOLS automated grading system Wordnet – an electronic lexical database – is used to supply alternative linking phrases for the existing relationships [11]. This approach is limited to those linking phrases that consist of a single verb. For each such a linking phrase, antonyms, synonyms and troponyms have been searched. When evaluated by a human grader, derivations of antonyms have been found to be always incorrect. In the meantime, derived synonyms and troponyms can be used for further identification of correct and incorrect propositions, but human grader is needed anyway to refine these sets; thus, the degree of automation is limited. The more propositions there are the more time has to be spent to refine the sets of alternative linking phrases.

Another attempt to extend the expert's CM is implemented in the rule-based evaluation system called Concept Mapping Tool (CMT) [12]. In this system the expansion of the expert's CM is performed manually by adding metarelations to the relationships. The metarelations can indicate one of the following three situations: equivalent relationships that can replace each other, inverse relationships and inexact replacements that are not incorrect, but also not ideal substitutes.

A scoring mechanism uses rules to evaluate propositions, and it assigns much more points for each matching proposition or correct inverse relation than for partially correct constructs, and the score is slightly reduced for each incorrect proposition. For the assessment of each proposition's correctness such components as the existence of a relation, a linking phrase

TABLE 1

GRAPH PATTERNS FOR CONCEPT MAP SCORING

Kind of patterns	Describe	Used	Points assigned	When scored
1st	Proposition components	To evaluate partly correct propositions	Full or part of score of completely correct proposition depending on correct components	If there is a complete match or a partial match
2nd	Hidden relationships	To evaluate propositions that are derivable from expert's CM (also for longer chains)	Less points than for those that an expert included in his/her CM	Only if matching relations are missing
3rd	Inverse relationships	To evaluate propositions that are directed in opposite way than those in expert's CM	Equal points than for those that an expert included in his/her CM	Only if matching relation is missing

and the direction of the arc are considered. Scores generated by CMT and a human evaluator strongly correlate, suggesting that scoring of the concept maps could be performed automatically. The main drawback of this approach is the extra work needed from an expert to manually expand the concept map.

IV. Graph patterns

All graph patterns are used to assess those constructs in a learner's CM that are not exact matches of constructs in the expert's CM. The first kind of graph patterns does not expand the expert's CM, while the second kind of graph patterns reveals hidden relationships and the third kind allows inference of inverse relationships. For comparison of the three kinds of graph patterns see Table 1. Graph patterns are compared considering four aspects: what kind of inexact matches they describe, which propositions they are used for, how many points are assigned and when the score is assigned for those propositions. In IKAS currently the first two kinds of graph patterns are implemented.

Graph patterns of the first kind describe 36 possible combinations of proposition components. Each proposition consists of 5 components that can be either correct or incorrect. Each of them has its weight when the score of the proposition is calculated. Proposition components and their respective weights are presence of the relationship in the learner's CM – 40%, a correct linking phrase provided for the relationship – 30%, a correct direction of the arc corresponding to the relationship – 15%, a correct type of a relationship (important or less important) – 10%, correct places of both concepts in the CM structure – 5% [7]. Not all of these components can be evaluated in every type of task. For example, places of concepts are always assumed to be correct in tasks where no CM structure is given and linking

TABLE 2

SET OF GRAPH PATTERNS FOR INFERENCE OF HIDDEN RELATIONSHIPS

	Rel.1	Rel. 2	Allowed	Rel. 3	Rule No.
1	Is a	Is a	Yes	Is a	R1
2	Is a	Part of	Yes	Cannot be specified*	-
3	Is a	Attribute	Yes	Cannot be specified*	-
4	Is a	Example	Yes	Is a	R2
5	Is a	Value	Yes	No extra relation**	R3
6	Is a	Kind of	Yes	Is a	R4
7	Part of	Is a	Yes	Part of	R5
8	Part of	Part of	Yes	Part of	R6
9	Part of	Attribute	Yes	Cannot be specified*	-
10	Part of	Example	Yes	Part of	R7
11	Part of	Value	Yes	No extra relation**	R8
12	Part of	Kind of	Yes	Part of	R9
13	Attribute	Value	Yes	No extra relation**	R10
14	Attribute	Example	Yes	No extra relation**	R11
15	Attribute	Attribute	Yes	No extra relation**	R12
16	Attribute	Part of	Yes	Cannot be specified*	-
17	Attribute	Is a	Yes	Cannot be specified*	-
18	Attribute	Kind of	Yes	Cannot be specified*	-
19	Example	Is a	Yes	No extra relation**	R13
20	Example	Part of	Yes	No extra relation**	R14
21	Example	Attribute	Yes	Cannot be specified*	-
22	Example	Example	No	-	R15
23	Example	Value	No	-	R16
24	Example	Kind of	Yes	No extra relation**	R17
25	Value	Any other except linguistic relation	No	-	R18
26	Kind of	Part of	Yes	Cannot be specified*	-
27	Kind of	Is a	Yes	Is a	R19
28	Kind of	Kind of	Yes	Is a	R20
29	Kind of	Example	Yes	Example	R21
30	Kind of	Attribute	Yes	Cannot be	-

				specified*	
31	Kind of	Value	Yes	Cannot be specified*	-

*Existence of an extra relationship depends on the semantics of concepts involved;

** No extra relation of considered 6 types.

phrases are assumed to be correct in those tasks where they are not used. Usage of the first kind of graph patterns allows assigning the score not only for completely correct propositions, but also for partly correct ones.

As this paper focuses more on a means for expansion of the expert’s CM graph patterns for the revelation of hidden relationships and inverse relationships are described more in next two sections.

V. Second kind of graph patterns

The second kind of graph patterns was first introduced in 2009 [13]. At that time it was assumed that every concept that represents an attribute of another concept is an indivisible unit. This assumption had to be abandoned because, when using IKAS for knowledge assessment in real domains, it appeared that it is necessary to represent situations where attributes consist of several parts as well as may create hierarchy-like structures [14].

TABLE 3

IF..THEN rules for the second kind of graph patterns

Rule No.	IF..THEN rule
R1	IF Relation(X, Y, “is a”) AND Relation (Y, Z, “is a”) THEN Relation (X, Z, “is a”)
R2	IF Relation (X, Y, “example”) AND Relation (Y, Z, “is a”) THEN Relation (X, Z, “is a”)
R3	IF Relation (X,Y, “is a“) AND Relation (X, Z, “value“) THEN NOT Relation (Y, Z, “value“) AND NOT Relation (Y, Z, “attribute“) AND NOT Relation (Z, Y, “is a“) AND NOT Relation (Z, Y, “kind of“) AND NOT Relation (Z, Y, “part of“) AND NOT Relation (Z, Y, “example“)
R4	IF Relation (X, Y, “kind of”) AND Relation (Y, Z, “is a”) THEN Relation (X, Z, “is a”)
R5	IF Relation(X, Y, “is a”) AND Relation(Y, Z, “part of”) THEN Relation(X, Z, “part of”)
R6	IF Relation(X, Y, “part of”) AND Relation(Y, Z, “part of”) THEN Relation(X, Z, “part of”)
R7	IF Relation(X, Y, “example”) AND Relation(Y, Z, “part of”) THEN Relation(X, Z, “part of”)
R8	IF Relation (X,Y, “part of”) AND Relation (X, Z, “value”) THEN NOT Relation (Y, Z, “value”) AND NOT Relation (Y, Z, “attribute”) AND NOT Relation (Z, Y, “is a”) AND NOT Relation (Z, Y, “kind of”) AND NOT Relation (Z, Y, “part of”) AND NOT Relation (Z, Y, “example”)
R9	IF Relation (X, Y, “kind of”) AND Relation (Y, Z, “part of”) THEN Relation (X, Z, “part of”)
R10	IF Relation (X,Y, “attribute”) AND Relation (Y, Z, “value”) THEN NOT Relation (X, Z, “value”) AND NOT Relation (X, Z, “attribute”) AND NOT Relation (Z, X, “is a”) AND NOT Relation (Z, X, “kind of”) AND NOT Relation (Z, X, “part of”) AND NOT Relation (Z, X, “example”)
R11	IF Relation (X,Y, “attribute”) AND Relation (Z, Y, “example”) THEN NOT Relation (X, Z, “attribute”) AND NOT Relation (X, Z, “value”) AND NOT Relation (Z, X, “is a”) AND NOT Relation (Z, X, “kind of”) AND NOT Relation (Z, X, “part of”) AND NOT Relation (Z, X, “example”)
R12	IF Relation (X,Y, “attribute”) AND Relation (Y, Z, “attribute”) THEN NOT Relation (X, Z, “attribute”) AND NOT Relation (X, Z, “value”) AND NOT Relation (Z, X, “is a”) AND NOT Relation (Z, X, “kind of”) AND NOT Relation (Z, X, “part of”) AND NOT Relation (Z, X, “example”)
R13	IF Relation(X, Y, “is a”) AND Relation (Y, Z, “example”) THEN NOT Relation (X, Z, “part of”) AND NOT Relation (X, Z, “is a”) AND NOT Relation (X, Z, “example”) AND NOT Relation (Z, X, “attribute”) AND NOT Relation (Z, X, “value”) AND NOT Relation (X, Z, “kind of”)
R14	IF Relation(X, Y, “part of”) AND Relation (Y, Z, “example”) THEN NOT Relation (X, Z, “part of”) AND NOT Relation (X, Z, “is a”) AND NOT Relation (X, Z, “example”) AND NOT Relation (Z, X, “attribute”) AND NOT Relation (Z, X, “value”) AND NOT Relation (X, Z, “kind of”)
R15	IF Relation (X, Y, “example”) THEN NOT Relation (Z, X, “example”)
R16	IF Relation (X, Y, “example”) THEN NOT Relation (X, Z, “value”)
R17	IF Relation (X, Y, “kind of”) AND Relation (Y, Z, “example”) THEN NOT Relation (X, Z, “part of”) AND NOT Relation (X, Z, “is a”) AND NOT Relation (X, Z, “example”) AND NOT Relation (Z, X, “attribute”) AND NOT Relation (Z, X, “value”) AND NOT Relation (X, Z, “kind of”)
R18	IF Relation (X, Y, “value”) THEN NOT Relation (Z, Y, “part of”) AND NOT Relation (Z, Y, “is a”) AND NOT Relation (Z, Y, “example”) AND NOT Relation (Y, Z, “attribute”) AND NOT Relation (Y, Z, “value”) AND NOT Relation (Z, Y, “kind of”)
R19	IF Relation (X, Y, “is a”) AND Relation (Y, Z, “kind of”) THEN Relation (X, Z, “is a”)
R20	IF Relation (X, Y, “kind of”) AND Relation (Y, Z, “kind of”) THEN Relation (X, Z, “is a”)
R21	IF Relation (X, Y, “example”) AND Relation (Y, Z, “kind of”) THEN Relation (X, Z, “example”)

A set of graph patterns of this kind was also influenced by the revision of direction of standard relationships. When creating the first version of this kind of graph patterns all six standard relations were decided to be directed from more specific concept to a broader one. The direction of relations “attribute” and “value” were changed because in other sources, for example [15], [16], [17], they were directed contrariwise. Also the teachers that use IKAS tend to direct the “attribute” relationship from an object to an attribute and the “value” relationship from a concept denoting attribute to its value. As a result of the mentioned changes, 14 patterns were changed, and five new patterns were added to the set. The number of patterns is now 31 (Table 2), and there are 21 IF..THEN rules associated with these patterns (Table 3).

In the IF..THEN rules every relation is described in the following form: *Relation (<concept_1>, <concept_2>, <relation_type>)*. “*Concept_1*” and “*concept_2*” are not particular concepts; these are parameters that indicate the direction of the relationship. “*Relation_type*” is one of the six types mentioned before.

Graph patterns of the second kind describe three kinds of situations [13]:

1. a combination of relationships that is forbidden,
2. a combination, which is allowed, but no extra relation can be added: (a) adding extra relation is forbidden, (b) analysis of concept semantics is needed to decide if there could be any extra relation,
3. a combination, which is allowed and from which a hidden relationship may be inferred.

Rules that are associated with the patterns that describe the first situation and the situation 2a are used to prevent the possibility that the expert by mistake includes forbidden constructs in his CM. The patterns that describe the third situation are used for expansion of the expert's CM.

When creating a CM, a teacher must decide which connections between concepts express important knowledge in the particular domain, and thus, should be included into CM and which express only marginal knowledge and can be omitted. There is a possibility to draw a link between almost any two concepts, even if they are not related (in this case an appropriate linking phrase could be "not related to"), but these connections are of a minimal significance. Only those propositions that describe important knowledge should be included in a CM. The second kind of graph patterns reveals relationships that are less important than those, which the teacher has included in the criterion CM; thus, the score for these relationships is lower. The learner receives no points for a hidden relationship if he/she has created both obligatory relationships (Rel. 1 and Rel 2.) and also their derivation (Rel 3) [7]. Points for a hidden relationship are assigned only if one or both obligatory relations are missing. Currently hidden relationships in IKAS are scored 1 point while important relationships are scored 5 points and less important ones – 2 points each.

VI. The Third kind of graph patterns

If two concepts are connected with a relationship, this implies also that there is an inverse relationship [15]. The inverse relationship will usually be of another type (Fig. 1b), although in some cases there could be relations of the same type in both directions (Fig. 1a). If a relationship between concepts is expressed using a verb, then a linking phrase for an inverse relationship can be derived using the passive voice.

The third kind of graph patterns describe the type of an inverse relation that can be used as a completely correct substitute for a relationship of a particular type in the expert's CM. For the standard relationships, such as "is-a", "part-of", there is a conventional direction of the link, but it does not mean that the opposite relationship is incorrect. These patterns describe situations where the learner has represented the same knowledge as an expert only in a different form; thus, the same amount of points should be assigned.

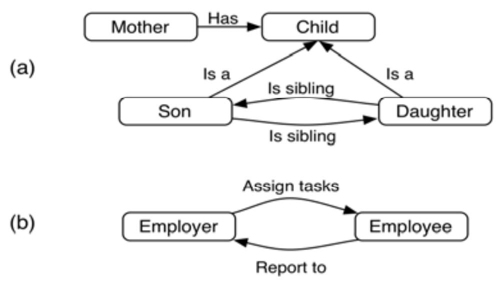

Fig. 1 Examples of reverse relationships

These patterns are created for each of six linking phrases that are currently used as predefined relation types, as well as for those 7 linking phrases that are suggested to be included as predefined types in IKAS (see Section 2).

Any graph patterns of the third kind are not yet implemented in IKAS because this set has been developed only recently. The problem of assessing inverse relations is not as crucial at the moment because there are no tasks in IKAS, where the learner would have to write LPs himself/herself. But as it is planned to expand the range of tasks by adding tasks where the learner will have to provide linking phrases and/or concept names instead of selecting them from a list, it will be important to be able to assess inverse relations as well [10].

All the graph patterns that allow inference of an inverse relation are described in the 4th table. In the first column the linking phrase of the expert's created relation is given, while the second column gives information about the direction of this relationship. The appropriate linking phrase for an inverse relation can be found in the third column. The corresponding IF..THEN rule for each pattern is in the rightmost column of the table. In these rules, relations are described in the same form as they are in the rules for the second kind of graph patterns.

TABLE 4

Set of patterns and their respective IF..THEN rules for inference of hidden relationships (Adapted from [10])

Relation type	Direction	Inverse relation	Corresponding IF..THEN rule
Is-a	Subclass-> Class	Subclass-is	IF Relation (X, Y, "is a") THEN Relation (Y, X, "subclass-is")
Kind-of	Subkind->Kind	Subkind-is	IF Relation (X, Y, "kind of") THEN Relation (Y, X, "subkind is")
Part-of	Part->Whole	Consists-of	IF Relation (X, Y, "part of") THEN Relation (Y, X, "consists of")
Example	Example-> Class	For-example	IF Relation (X, Y, "example") THEN Relation (Y, X, "for example")
Attribute	Object-> Property	Characterizes	IF Relation (X, Y, "attribute") THEN Relation (Y, X, "characterizes")
Value	Property-> Value	Value-for	IF Relation (X, Y, "value") THEN Relation (Y, X, "value-for")
Before	First event-> Second event	After	IF Relation (X, Y, "before") THEN Relation (Y, X, "after")

Located-at	Object/Event-> Location	Location-of	IF Relation (X, Y, "located at") THEN Relation (Y, X, "location of")
Cause	Cause-> Effect	Caused-by	IF Relation (X, Y, "cause") THEN Relation (Y, X, "caused by")
Effect	Agent->Patient	Affected-by	IF Relation (X, Y, "effect") THEN Relation (Y, X, "affected by")
Has-function	Object-> Function	Function-of	IF Relation (X, Y, "has function") THEN Relation (Y, X, "function of")
Made-of	Object-> Material	Used-to-make	IF Relation (X, Y, "made of") THEN Relation (Y, X, "used to make")
Possesses (Has)	Possessor-> Possession	Possessed-by	IF Relation (X, Y, "possesses") THEN Relation (Y, X, "possessed by")

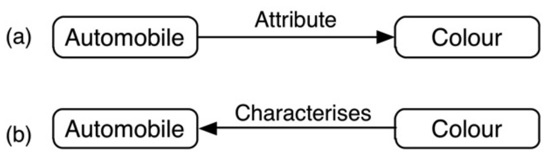

Fig.2. Example of a correct reverse relationship

There are two degrees of task difficulty where linking phrases are not used at all. Nevertheless, inverse and hidden relationships can be scored in those tasks too, because the expert's CM always contains linking phrases, so it is possible to derive the hidden relationships and reverse relationships for expansion of the expert's CM and score them.

Usage of graph patterns for inference of inverse relationships differs from the usage of the first kind of graph patterns. Let us compare the score of a CM fragment (Fig. 2), using only the first kind of graph patterns and using the third kind of graph patterns. Figure 2a represents a fragment of an expert's CM, and Figure 2b represents a fragment of a learner's created CM. If the learner's construct is scored using only the first kind of graph patterns, he/she would receive only 55% of the score (if it is a "construct-a-map-task" or a "fill-in-the-map" task, and concepts are placed in correct places into the given CM structure) that is 2.75 points in case of an important relationship and 1.1 point in case of a less important relationship. When only these patterns are used, an inverse proposition is considered to have two incorrect components: a linking phrase and an arc direction. If graph patterns of the third kind are used to expand an expert's CM by adding inverse relations before comparing it to the learner's CM, 5 or 2 points would be received in case of an important or less important relationship accordingly.

VII. Conclusions and Future Work

Although there are other attempts to add inverse relations and inexact matches to an expert's CM, usage of graph patterns to do it automatically is a novel approach that helps to minimize a teacher's workload and allows assessing a learner's knowledge level more precisely. Usage of graph patterns for CM scoring purposes allows more freedom for a learner, expressing his/her knowledge structure, instead of forcing to use certain names and structures. There are mechanisms for a partly automated addition of equivalent relationships described in literature, while in IKAS currently equivalent relations are assigned manually by an expert adding possible synonyms for linking phrases.

Graph patterns are an important part of an automated CM scoring mechanism, but there are still rather many open questions regarding a scoring mechanism that could automatically assess a learner's knowledge giving rich information on a learner's knowledge level. A few of them are as follows: should multiple relations between two concepts be considered? Are there other subgraphs (like trees, cycles, paths and stars) that have additional meaning in aggregate? Are there more linking phrases that should be defined as standard relation types? How to assess such a situation when a learner has created such a concept that corresponds to a proposition (2 concepts and a relationship) in the expert's CM? It is possible that some of these questions could be answered introducing yet another kind of graph patterns.

Patterns for addition of inverse relations are currently defined only conceptually; thus, they should be implemented and evaluated experimentally using them for expansion of CMs of various domains.

Acknowledgement

This research has been supported by the European Social Fund within the project "Support for the Implementation of Doctoral Studies at Riga Technical University".

References

[1] S. Waterhouse *The Power of E-learning. The Essential Guide for Teaching in the Digital Age.*Allyn& Bacon, 2004, p. 228.

[2] J.D. Novak *Learning, Creating, and Using Knowledge: Concept Maps as Facilitative Tools in Schools and Corporations*. [Kindle e-book], LEA, 2008, Available: Amazon.

[3] J. Seale, *Using CAA to support student learning*, [Online]. Available: http://www.alt.ac.uk/sites/default/files/assets_editor_uploads/documents/eln004.pdf [Accessed: Oct.1, 2011].

[4] M.A. Ruiz-Primo, R.J. Shavelson, "Problems and Issues in the Use of Concept Maps in Science Assessment," In: Journal of Research in Science Teaching, vol.33, no.6, pp. 569—600, 1996.

[5] J.D. Novak, A.J Canas. "The Theory Underlying Concept Maps and How to Construct and Use Them," January 2006. [Online] Available :http://cmap.ihmc.us/publications/researchpapers/theorycmaps/theoryunderlyingconceptmaps.htm, [Accessed: Oct. 1, 2011].

[6] "Cognitive Theories of Learning." [Online]. Available: http://personal.psu.edu/users/w/x/wxh139/cognitive_1.htm[Accessed: Oct. 1, 2011].

[7] A. Anohina-Naumeca, J. Grundspenkis, M. Strautmane, "The Concept Map Based Assessment System: Functional Capabilities, Evolution, and Experimental Results." International Journal of Continuing Engineering Education and Life-Long Learning, Vol. 21, No. 4, pp.308-327, 2011.

[8] J. Grundspenkis, "Concept map based intelligent knowledge assessment system: Experience of development and practical use" in Multiple Perspectives on Problem Solving and Learning in Digital Age, D. Ifenthaler et al., Eds., 2011, Ch. 12, pp. 179-198.

[9] R. Lukašenko, A .Anohina-Naumeca, M. Viļķelis, J. Grundspeņķis, "Feedback in the Concept Map Based Intelligent Knowledge Assessment System," Scientific Journal of RTU, 5 series., "Computer science" – vol. 43, 2010, pp 17-26.

[10] M. Strautmane, *"Expansion of a Set of Graph Patterns for IKAS"*, in Proc. INTEL-EDU '11 workshop, Riga, Latvia,2011 (accepted for publishing) .

[11] S. H. Harrison, et al. *"C-TOOLS Automated Grading for Online Concept MAps Works Well with a Little Help from "Wordnet""*, in Proc. 1st International Conf. Concept Mapping. A. J. Canas, J. D. Novak, F.

M. Gonzalez, Eds. September 14- 17, 2004, Pamplona, Spain, pp.211-214.

[12] B.E. Cline, C.C. Brewster, R.D. Fell, *"A rule-based system for automatically evaluating student concept maps",* in Expert systems with applications vol. 37, 2010, pp: 2282-2291.

[13] J. Grundspenkis, M.Strautmane, "Usage of Graph Patterns form Knowledge Assessment Based on Concept Maps". Scientific Journal of RTU, 5. Series, "Computer science", vol. 38, 2009, pp 60-71.

[14] M. Strautmane *"Īpašībasaprakstošojēdzienusaistībaarcitiemjēdzieniemkonceptukartēs",* in 51. RTU studentuzinātniskāsuntehniskāskonferencesrakstukrājums-lietišķāsdatorsistēmas, Riga, Latvia, April 23, 2010. pp: 77-82 (in Latvian).

[15] "Semantic Networks". [Online]. Available: http://www.duke.edu/~mccann/mwb/15semnet.htm [Accessed: Oct. 1, 2011].

[16] J.F. Sowa "Semantic Networks". [Online]. Available: http://www.jfsowa.com/pubs/semnet.htm [Accessed Oct. 1, 2011].

[17] J.R. McClure, P.E. Bell, "Effects of an environmental Education-related STS Approach Instruction on Cognitive Structures of Preservice Science Teachers", Pennsylvania State University, University Park. Coll. of Education, USA, Tech. Rep. 1990. (ERIC Document Reproduction Service No. ED 341 582)

Maija Strautmane is a PhD student of Computer Systems at Riga Technical University in Latvia. She obtained her Master's Degree in Computer Systems from the mentioned institution in 2010. Her thesis was related to concept map scoring methods that would provide a valid and reliable evaluation of learner's knowledge.

Maija Strautmane currently works as an Assistant Lecturer at the Institute of Applied Computer Systems, Riga Technical University, Latvia. Previously she worked there as a Laboratory Assistant. Since 2007 she has participated in several projects as a part of Intellectual and adaptive concept map based knowledge assessment system IKAS development group.

E-mail: maija.strautmane@rtu.lv