Concept Maps: Theory, Methodology, Technology
Proc. of the First Int. Conference on Concept Mapping
Pamplona, Spain 2004

LINKING PHRASES IN CONCEPT MAPS: A STUDY ON THE NATURE OF INCLUSIVITY

Júlio Valente da Costa Jr, Francisco Edson Lopes da Rocha & Eloi Luiz Favero, Universidade Federal do Pará, Brasil
Email: felr@ufpa.br, www.inf.ufpa.br/~edson

Abstract. Computerized analysis of a concept map (CM) syntactic and semantic characteristics can become a complex task, whose results can be doubtful if the map contains either graphical or textual ambiguities. In the specific case of using CMs in education, the use of the results of automatic structural analysis of a map in learning assessment implies the existence of entirely computable maps, capable of coping with the inherently ambiguous nature of the language used in their construction, which results in imprecise concepts and linking phrases. This imprecision hinders initiatives toward computer processing of CMs. In this paper, we present a strategy to minimize the imprecision of linking phrases, by analyzing the nature of inclusivity in hierarchies of concepts in CMs. We present a formal extensible notation capable of distinguishing among the subtleties of these hierarchies. Based on this formalization, it is possible not only to determine precisely the dimension of inclusivity specified in a proposition, but also to define methods capable of determining the semantic distance between propositions in CMs.

1 Introduction

During the process of learning, learners construct a reality or interpretation of reality based upon their perceptions. Traditional conceptions of learning emphasize the object of our knowing rather than the process of coming to know what is actually learned. Constructivism, on the other hand, focus on the mental processes inherent in the construction of meaning. These mental processes are highly dependent on the learner's prior knowledge, current mental structures, and existing beliefs (Jonassen et al., 1993).

"Concept maps" (Novak & Gowin, 1984) and other knowledge representation tools – of which the most widely known are "Semantic networks" (Fisher et al., 1990) and "Mind maps" (Buzan & Buzan, 1993) – have been used in computer-based applications and paper and pencil applications, in accordance with the learning models proposed by constructivist theorists such as Ausubel, Vygotsky, and von Glasersfeld. They support the activities of learning, teaching, research, intellectual analysis, and organization of knowledge resources, by mimicking the workings of the brain, especially working memory and long-term memory (Fisher et al., 2000).

Concept mapping is a process of meaning construction. The concept maps (CMs) that result from this process are diagrams – usually bi-dimensional – that illustrate relationships between two or more concepts. Concepts can be defined as regularities perceived in objects, events, situations, or properties (Novak & Gowin, 1984). Another common definition states that concepts are objects, facts, situations, or properties that possess common criterial attributes and are represented by the same symbol (Ausubel, 2000). A proposition is a relationship or association between two or more concepts.

CMs are based in three fundamental characteristics: hierarchical structure, progressive differentiation, and integrative reconciliation (Novak & Gowin, 1984). The hierarchical structure of a CM is based on the concept of **inclusivity**: each concept inserted in a map has a level of inclusivity somehow comparable to the levels of the concepts already in the map. This is not necessarily related to the physical containment sense of the word "inclusivity". In the context of CMs, inclusivity is a concept of epistemological nature, fundamental for the construction of meaning. In CMs, the idea of inclusivity alludes to the very nature of hierarchies and their expressiveness. A more inclusive concept is one that, according to the learning task in progress, can be considered a superordinate concept.

Linking phrases used to label propositions can be considered values assigned to binary relations that exist between concepts. These values can be organized in hierarchies of **super-types**, according to the semantic relationship (**dimension**) resulting from their use between two concepts, as in Fisher (1988). This categorization is important for the automatic analysis of maps and for the determination of levels of similarity between propositions. For example, in the propositions <CAR has WHEEL> and <CAR depends on FUEL> the linking phrases <has> and <depends on> are instances of the super-types **Partition** and **Dependency**. They imply partition and dependency dimensions to the relations between the concepts. Figure 1 illustrates a concept map in which <HUMAN LEARNING> is the concept of highest inclusivity degree. It is related to the concepts <COGNITIVE LEARNING>, <AFFECTIVE LEARNING> and <PSYCHOMOTOR LEARNING> according to a **Classification** dimension.

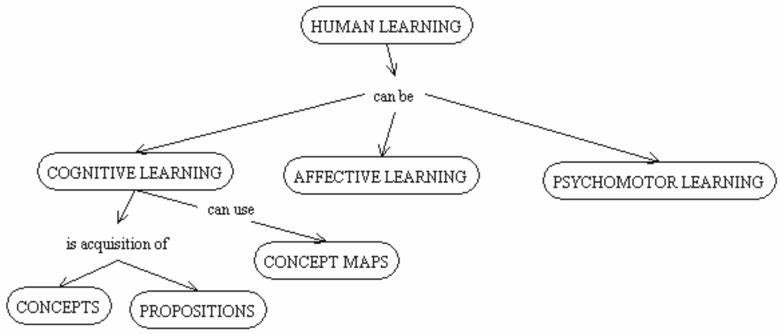

Figure 1. Concept map about human learning

Linking phrases in propositions not only determine the semantic dimension in which two concepts are related from a hierarchical perspective. They are also fundamental in the processes of **progressive differentiation** and **integrative reconciliation**, because they limit the scope of the differentiation or reconciliation in progress.

Progressive differentiation is the process of meaningful learning in which learners increase the degree of elaboration of a concept as they increase their knowledge about it (Ausubel, 2000). In order to detect progressive differentiation in a concept map, it is necessary to observe if more inclusive concepts are related to less inclusive concepts in certain dimensions. In Figure 1, the concept <COGNITIVE LEARNING> is progressively differentiated.

Integrative reconciliation, on the other hand, occurs when the learner identifies dimensions of relationships between components not previously connected. In **superordinate integrative reconciliation**, the learner identifies a more inclusive concept not initially present in the map or not initially connected to the less inclusive concepts. **Combinatorial integrative reconciliation** happens when the learner perceives dimensions of relationships between concepts that, according to the learning task, are not part of an identifiable hierarchy. The learner does not identify a more inclusive concept, but discerns the need of relating concepts present in different branches of the map.

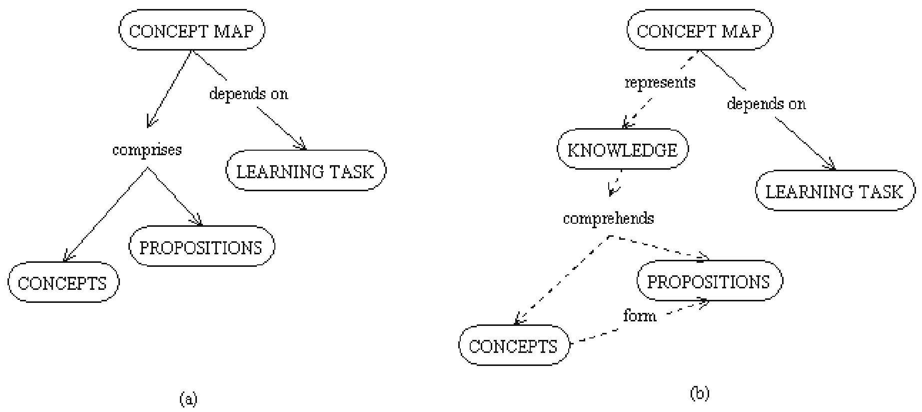

Figure 2. Progressive differentiation and integrative reconciliation in concept maps

Figure 2(a) illustrates the progressive differentiation of the concept <CONCEPT MAP>. Less inclusive concepts are related to <CONCEPT MAP> in a **Partition** dimension and in a **Dependency** dimension. Figure 2(b) illustrates two situations of integrative reconciliation: superordinate reconciliation (among the new concept <KNOWLEDGE> and its subordinates <CONCEPTS> and <PROPOSITIONS>) and combinatorial reconciliation (in the binary relation <CONCEPTS form PROPOSITIONS>.

This analysis provides an insight on the importance of linking phrases in concept maps, as well as their classification in the semantic dimensions they represent. The task of linking concepts into propositions is the most critical step in constructing CMs (Jonassen, 1996; Cañas, 2003), because this is the moment in which meaning is being constructed and refined. The objective of this article is to shed some light on how linking phrases can be automatically computed, not only with respect to the dimensions they represent, but also on how their meanings can be compared. It contains five sections, including this introduction. In section 2, we present some simple cases of ambiguity in CMs. In section 3, we present an excerpt of the computable formalization (grammar) for describing inclusivity in CMs. Section 4 presents a scenario of use: the CMTool learning environment, in which the grammar is used to determine semantic distances between two given propositions. We conclude the article in section 5.

2 Ambiguity in CMs

CMs are simplified representations of a person's cognitive structure (Novak, 1984; Peña et al., 1996; McAleese et al., 1999). As mentioned in McAleese et al. (1999), there is strong evidence that the representations we use in thinking are largely not language based or language dependent. Rather, people think with mental imagery and skeletal representations of ideas. These concepts, maintained abstractly in the cognitive structure, are highly interconnected, forming propositions that express new dimensions of the original concepts. However, ambiguous natural (human) language is the main tool in the process of concept mapping. In CMs, both concepts and propositions are represented in natural language and, as a result, are subject to ambiguity.

The ambiguity resulting from the process of concept mapping makes it more difficult to automatically analyze and assess learning in CMs. Although this ambiguity can occur in concepts or propositions, we focus on the latter, because of the importance of the process of connecting concepts in concept mapping. For more information on disambiguating concepts in CMs, refer to Cañas et al. (2003), in which a **disambiguator algorithm** based on information from WordNet (Fellbaum, 1998) is presented. Other WordNet-based initiatives in determining word sense can be found in Li et al. (1995) and Nastase & Szpakowics (2001).

The process of connecting concepts via linking phrases is critical because of the effort exerted by the learner in trying to determine exactly the nature of the relationship between the concepts (Jonassen, 1996). Figure 3 illustrates an example of propositional ambiguity, considering the concepts are unambiguously determined.

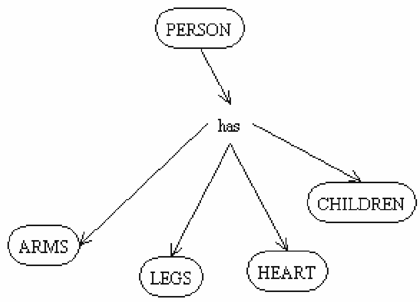

Figure 3. CM with ambiguous use of linking phrase (<has>)

In the proposition <PERSON has ARM> the dimension the learner probably wants to express is a **Partition** dimension connecting and expanding the concepts <PERSON> and <ARM>. On the other hand, in the proposition <PERSON has CHILDREN>, the underlying idea is not of a partition dimension, but rather of an "offspring" or "tutoring" dimension. A computerized analysis of such a CM would have to face this ambiguity and would not be able to assess the students understanding of the learning task straightforwardly. In the example given in Figure 3, the nature (super-type) of the linking phrase can be inferred from the concepts being connected and from the context (learning task) under analysis. This can be done by a disambiguator algorithm capable of informing (e.g. through usage probability) the dimension that best fits the proposition. However, more complex CMs are hard to be analyzed if the analysis is not supported by some kind of systematization of possible dimensions of inclusivity. Another example of ambiguous use of a linking phrase, this time in a more complex domain, is given in Figure 4.

The propositions <ARTIFICIAL NEURAL NETWORK consists of LAYERS OF NEURONS> and <ARTIFICIAL NEURAL NETWORK consists of MATHEMATICALLY MODELED NEURONS> do not express exactly the same idea: the basic foundation element of artificial neural networks are neurons, which are usually organized in layers. Thus, a traditional computerized system designed to assess students understanding of the topic would not be able to discern whether the student understands that the dimension of the relationship between neural networks and neurons is a **Partition.Aggregation** dimension and between neural networks and layers is a **Partition.Organization** dimension. In this case, a disambiguator mechanism could use contextual information (extracted from nearby concepts or from the learning task underway) to derive the correct meaning of the linking phrase. Another possible solution would be to request further explanation from the student drawing such a map. This further explanation could be given by determining explicitly the nature (super-type) of the linking phrase. In both cases, however, an initial step is to create a categorization of the linking phrases, either for use by the disambiguator, or for student explanation.

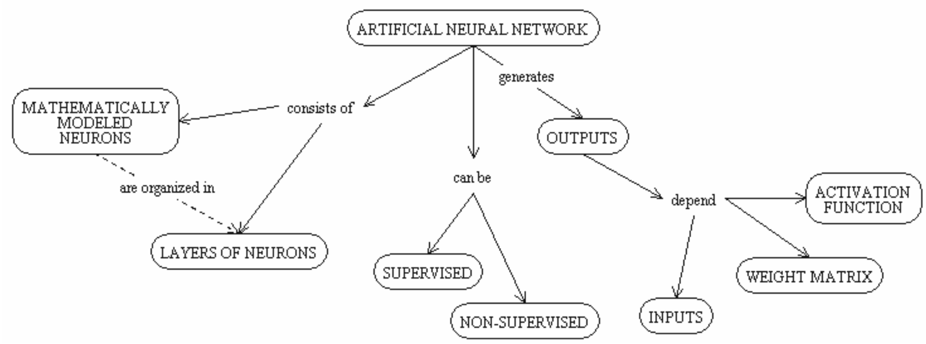

Figure 4. CM with ambiguous use of linking phrase (<consists of>)

3 An EBNF Grammar Definition of Inclusivity in CMs

3.1 The EBNF notation

In order to make it possible to create a categorization of linking phrases capable of being parsed by a computer system, and also capable of being extended as needed, it is necessary to use a notation sufficiently formal to allow for parsing and calculations of distances between two points in the categorization. The EBNF notation defined in the ISO/IEC 14977 standard (ISO/IEC 14977:1996(E), 1996) is a formal mathematical notation used to describe a language. It consists of a collection of rules (**productions)** collectively called a **grammar**. It is widely used in Computer Science to formally define the grammar of a programming language, operating system commands, and other types of computer input.

In EBNF, each production consists of a non-terminal symbol and an EBNF expression separated by an equal sign and terminated with a semicolon. The non-terminal symbol is called **meta-identifier** (a syntactic constant denoted by an English word) and the EBNF expression is its definition. The EBNF expression is composed of zero or more terminal symbols, non-terminal symbols, and other metasymbols. EBNF cannot capture non-context-free grammars, so an EBNF grammar may need to be augmented with notes on semantics, in free-form English (although other more formal approaches may be used). An exemplifying grammar defining the mathematical set of integers is illustrated in Figure 5.

```
integer = unsigned integer | ‘+’,unsigned integer  |  ‘-’ unsigned integer;
unsigned integer = digit | unsigned integer, digit;
digit = ‘0’ | ‘1’ | ‘2’ | ‘3’ | ‘4’ | ‘5’ | ‘6’ | ‘7’ | ‘8’ | ‘9’;
```

Figure 5. Example definition using EBNF

3.2 The Inclusivity Grammar

Some attempts to classify and categorize binary relations connecting two concepts (or ontological nodes) exist in the knowledge representation literature. Yudelson et al. (2003) list two initiatives in this direction, when designing the knowledge ontology present in their model of adaptive on-line education system: The IEEE Learning Objects Metadata (LOM) and the Multibook project. The IEEE LOM Model specifies the syntax and semantics of learning objects metadata (Fischer, 2001). These objects can be considered concepts in an N-dimensional map, in which a new dimension is defined when a new type of object is used (examples of objects include multimedia content, instructional content, instructional software, and so on).

Multibook is an adaptive hypermedia system used to teach multimedia technology (Multibook, 2004). Multibook’s knowledge base works with two different spaces: the ConceptSpace and the MediaBrickSpace. The ConceptSpace is an ontology, in which each concept is defined in terms of keywords that are used to sketch a lesson. The MediaBrickSpace contains the actual content to be displayed (text, images, audio, video, animation) (Fisher, 2001). The elements in these two spaces are connected via binary relations similar to the ones found in

CMs. These semantic relations are not intended to be complete and need to be redesigned in case the model is used in another domain.

Although these initiatives are not directly connected to the concept mapping process, they list some very common linking phrases found in CMs. A more complete categorization of binary relations, collected directly from linking phrases used by learners in CMs, was defined by Kathleen Fisher (Jonassen, 1996), resulting from her long experience with semantic networks, using the SemNet software (Fisher, 1992). This categorization is the basis of our grammar, because it is a fairly comprehensive list of link types that may be used to connect concepts during the mapping process. The resulting EBNF grammar is capable of determining the possible dimensions of linking phrases in propositions. The grammar itself can be graphically seen as a CM, as illustrated in Figure 6.

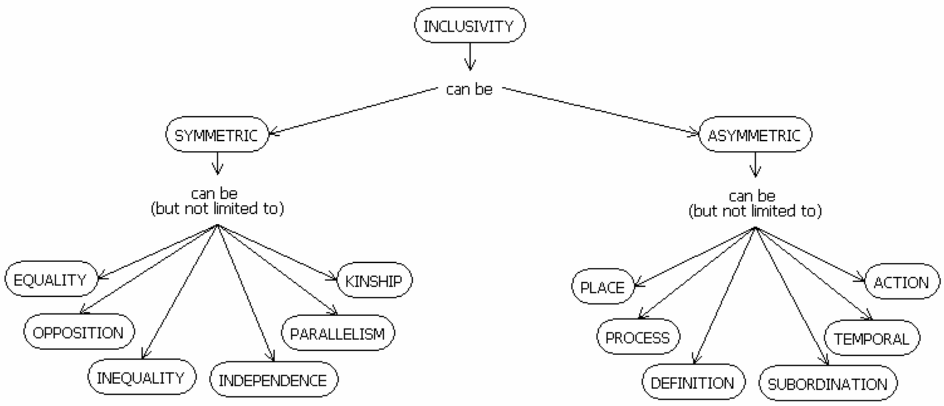

Figure 6. CM about the inclusivity grammar

Each of the non-terminal symbols in the grammar (concepts in Figure 6) can be determined in terms of terminal symbols that encompass its dimension, or can be further elaborated, by the use of recursive non-terminal symbols. An example is provided in Table 1, in which the non-terminal **Definition** is detailed.

Symbol	**Type**
Analytical	Non-terminal
Synthetical	Non-terminal
'is described by'	Terminal
'is defined by'	Terminal
'describes'	Terminal
'defines'	Terminal

Table 1. Description of the non-terminal Definition

According to the description above, excerpts of the EBNF grammar for inclusivity can be seen in Figure 7. This grammar is in constant evolution and is by no means complete, because it is not possible to completely describe all the permissible relations in a CM. It is, however, an attempt to classify the most commonly used binary relations, yet allowing for improvements and extensions (depending on the domain under analysis). Although the grammar alone can be the basis of an automated CM parser, it can also have its terminal symbols connected to a lexicon capable of linking words with synonyms, like the **synsets** in WordNet (Fellbaum, 1998). As a result, the capabilities of the parser would be greatly enhanced. It would be able to analyze the expressions in the grammar and also various synonymous expressions in the lexicon.

```
Inclusivity= (Symmetric | Asymmetric);
    Symmetric= Opposition | Equality | Inequality | Kinship | Independence | Parallelism;
        Opposition=(Logical | Physical);
            Logical='does the opposite of' | 'is opposite of' | 'is contrary' | 'has antonym' | 'is opposed
            to';
            Physical='is opposite';
        Equality=(Logical |Physical);
            Logical='equals' | 'is same as' | 'has synonym';
            Physical='is in same place';
    (…)
    Asymmetric=Definition | Subordination | Place | Temporal | Action | Process;
        Definition=Analytical | Synthetical | 'is described by' | 'is defined by'| 'is denoted by' |
```

‘describes’ | ‘defines’| ‘denotes’;
Analytical=(Partition | Characterization);
Partition=(Aggregation | Composition | Organization);
Aggregation= (Temporal | Non-Temporal);
Temporal= ‘has step’ | ‘has stage | ‘is step of’ | ‘is stage of’ ;
Non-Temporal=‘has part’ | ‘has piece’ | ‘contains’ | ‘is part of’ | (…)
Composition=(Temporal | Non-Temporal);
Temporal= ‘has’ | ‘has step’ | ‘has stage | ‘is step of’ | ‘is stage of’ ;
Non-Temporal= ‘has part’ | ‘has piece’ | ‘contains’ | ‘is part of’ | (…)
Organization= ‘is organized in’;
(…)

Figure 6. Excerpts of the EBNF Grammar for Inclusivity (indentation is used to facilitate reading)

Some points about the grammar are important: 1) other classifications are possible and may depend on the domain under analysis (our attempt is to classify the most generic relations); 2) although the grammar is self-explanatory in its majority, it may be confusing at first to discern among Aggregation, Composition, and Organization. Aggregation is a whole-part dimension in which the whole still exists when deprived of one of its parts. Composition is a whole-part dimension in which the whole does not exist by itself when deprived of one of its parts. Organization does not describe elementary units of a whole, but rather organizational units, that exist only for the sake of explaining or organizing the whole; 3) expressions like ‘causes’ and ‘is caused by’ permeate the grammar. Although they differ from each other (one is in the active voice, the other is in the passive voice), one can be easily transformed into the other. Since they represent the same semantic notion, we decided to classify them in the same super-type, although more detailed grammars could separate them.

4 Scenario of Use: CMTool

The categorization of linking phrases in the form described in this article is being implemented in the CMTool environment (Rocha & Favero, 2004). Figure 7 presents CMTool’s block diagram, whose architecture includes five modules and a repository. The ontology, the genetic algorithm (GA) and the assessor work together to produce an assessment of a CM. The CM editor implements a visual language sufficiently expressive to draw CMs that can be analyzed automatically without risk of ambiguity. The administrator controls the access to the environment. The repository stores CMs, ontologies and user information.

CMTool considers learning assessment as an adaptive and evolutionary problem. Therefore, instead of comparing the learner’s CM (CM_{ap}) to a reference CM (CM_r), the environment contrasts CM_{ap} with a collection of CMs generated by the GA for the learning task underway (the collection is a population and each CM in the population is an individual). Based on a function centered in the categorization of linking phrases, the assessor calculates the semantic distance (proximity) from CM_{ap} to each of the individuals in the population. This is a process that enriches learning assessment, as it is possible to present to the learner all the possible alternatives to CM_{ap}, according to what is registered in the learning task ontology. An example of this process is summarized below. For a more detailed discussion on how CMTool works, refer to Rocha et al. (2004).

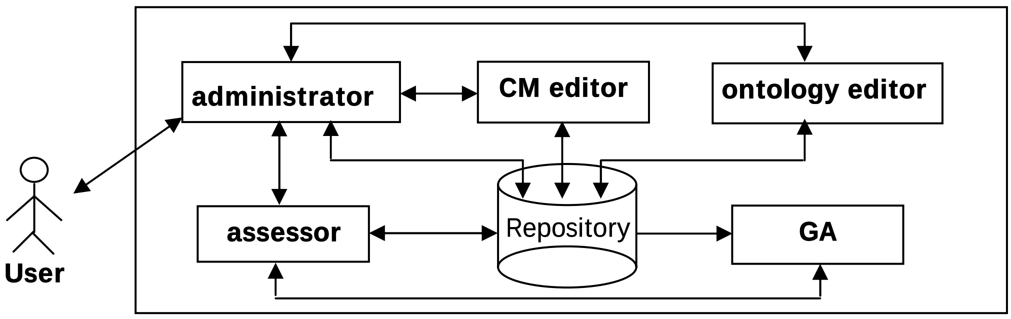

Figure 7. Architecture of the CMTool environment

Consider the learning task is to study *“the water cycle”* and consider that the domain ontology for this task – kept in the repository and entered via the ontology editor – can be summarized as follows: 1) concepts: WATER CYCLE, EVAPORATION, CONDENSATION, PRECIPITATION; 2) relations: a) temporal type: ‘has phase’,‘has stage’,‘precedes’,‘comes before’,‘is phase in’,‘is stage of’,‘succeeds’,‘comes next’; 3) binary relations: <WATER CYCLE,r_1,CONDENSATION>, <WATER CYCLE,r_2,EVAPORATION>, <WATER CYCLE,

r_3,PRECIPITATION>, <EVAPORATION,r_4,CONDENSATION>, <CONDENSATION,r_5,PRECIPITATION>; 4) values of r_1, r_2, r_3: 'has phase', 'has stage'; 5) values of r_4, r_5: 'precedes', 'comes before'.

With this information we can determine all the valid propositions in this context, as well as the semantic distances between them, as follows. Valid propositions: $\{p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{10}\}$ = {<WATER CYCLE, has phase, CONDENSATION>, <WATER CYCLE, has stage, CONDENSATION>, <WATER CYCLE, has phase, EVAPORATION>, <WATER CYCLE, has stage, EVAPORATION>, <WATER CYCLE, has phase, PRECIPITATION>, <WATER CYCLE, has stage, PRECIPITATION>, <EVAPORATION, precedes, CONDENSATION>, <EVAPORATIION, comes before, CONDENSATION>, <CONDENSATION, precedes, PRECIPITATION>, <CONDENSATION, comes before, PRECIPITATION>}; semantic distances between the propositions: a) $d_p(p_1, p_2) = d_p(p_3, p_4) = d_p(p_5, p_6) = d_p(p_7, p_8) = d_p(p_9, p_{10}) = 0$; b) $d_p(p_1, p_i) = \infty$, $1 \le i \le 10$, $i \ne 2$; c) $d_p(p_3, p_i) = \infty$, $1 \le i \le 10$, $i \ne 4$; d) $d_p(p_5, p_i) = \infty$, $1 \le i \le 10$, $i \ne 6$; e) $d_p(p_7, p_i) = \infty$, $1 \le i \le 10$, $i \ne 8$; f) $d_p(p_9, p_i) = \infty$, $1 \le i \le 10$, $i \ne 10$.

Semantic distances between propositions are symmetric, i.e. ($d_p(p_i, p_j) = d_p(p_j, p_i)$). The distance $d_p(p_i, p_j) = 0$ means that p_i and p_j are semantically equivalent, while $d_p(p_i, p_j) = \infty$ means that p_i and p_j are semantically incomparable. The function implemented also admits $d_p(p_i, p_j) = 1$ when p_i and p_j are semantically approximate (same binary relation, but the value assigned to the relation is not meaningful in the context, according to the ontology). The semantic distance between CMs is defined as a function of the semantic distances of their comparable propositions (the ones that contain the same concepts). For example, d_p(<WATER CYCLE, has phase, CONDENSATION>, <WATER CYCLE, has stage, CONDENSATION>) = 0, because <has phase> and <has stage> are both included in the **Temporal** type that describes the relation <WATER CYCLE, r_1, CONDENSATION>. Considering CM_1 = {<WATER CYCLE, has phase, CONDENSATION>, <WATER CYCLE, has phase, EVAPORATION>} and CM_2 = {<WATER CYCLE, has phase, CONDENSATION>, <WATER CYCLE, has stage, EVAPORATION>}, $d_m(CM_1, CM_2) = 0$, because max((d_p(<WATER CYCLE, has phase, CONDENSATION>, <WATER CYCLE, has phase, EVAPORATION>), d_p(<WATER CYCLE, has phase, CONDENSATION>, <WATER CYCLE, has stage, EVAPORATION>)) = 0, where *max* is the biggest value in the argument list of the function.

5 Conclusions

In this article we show how to automatically compute linking phrases, analyzing the nature of inclusivity in CMs. The objective of the study is to define a disambiguator mechanism capable of interpreting automatically, univocally, the binary relations between concepts in a CM. We have used the EBNF notation to formalize a free-context grammar that describes a hierarchy based on the semantic inclusion concept.

The meaningful learning theory is a viable form of constructivism if it is mediated by a tool like concept map. Nevertheless, constructivist learning is inherently adaptive (idiosyncratic) and evolutionary. It is adaptive because there are several ways in which humans can construct the same knowledge (Novak, 1998) and it is evolutionary because new knowledge, learned meaningfully, is always anchored in previous knowledge (Ausubel, 2000).

The experiments we have carried out at our education institution have showed us that we need more than a reference CM (one constructed by a specialist, for example) to assess learners' CMs, considering this idiosyncratic character of meaningful learning. That is the reason why we have concentrated on mechanisms capable of generating collections of CMs to represent this variability in learning, and capable of storing the knowledge that can be mapped in the context of a learning task. Our proposal of CM assessment involves the use of domain ontologies and machine learning via genetic algorithms (GAs). In CMTool (Section 4), ontologies are mechanisms that store knowledge in the form of concepts and binary relations between them. They also store the inclusivity grammar described in this paper, which is used by the GA to analyze and determine the meaning of generic binary relations. The inclusivity grammar can also be used to compare and contrast CMs. Our first version of this grammar, developed in Prolog, is currently being converted to Java.

Ausubel (2000) says that the cognitive structure is characterized by factors (or variables) that need to be enhanced for the improvement of education. Therefore, more research needs to be carried out to predict or measure stability, clarity, and discriminability of the cognitive structure. Our results are a step towards this direction.

6 References

Ausubel, D. P. (2000). *The Acquisition and Retention of Knowledge*. New York: Kluwer Academic Publishers.

Buzan, T. & Buzan, B. (1993). *The Mind Map Book: How to Use Radiant Thinking to Maximize your Brain's Untapped Potential.* Toronto: Plume Books (Penguin).

Cañas, A. J.; Valerio, A.; Lalinde-Pulido, J.; Carvalho, M.; Arguedas, M. Using WordNet for Word Sense Disambiguation to Support Concept Map Construction. *SPIRE 2003 – 10th International Symposium on String Processing and Information Retrieval*, October 2003, Manaus, Brazil.

Fellbaum, C. ed., *WordNet – An Electronic Lexical Database*, MIT Press, 1998.

Fischer, S. (2001). Course and Exercise Sequencing Using Metadata in Adaptive Hypermedia Learning Systems. *ACM Journal of Educational Resources in Computing 1(1).*

Fisher, K. M. (1988). Relations Used in Student-Generated Knowledge Representations. *Symposium on Student Understanding in Science*, American Educational Research Association, New Orleans.

Fisher, K. M., Faletti, J., Patterson. H. A., Thornton, R., Lipson, J., & Spring, C. (1990). Computer-based concept mapping: SemNet software - a tool for describing knowledge networks. *Journal of College Science Teaching, 19 (6).*

Fisher, K. M. (1992). SemNet: A Tool for Personal Knowledge Construction. In P. Kommers, D. Jonassen, & T. Mayes (Eds.), *Cognitive Tools for Learning*. Berlin: Springer-Verlag.

Fisher, K. M., Wandersee, J. H., & Wideman, G. (2000). Enhancing cognitive skills for meaningful understanding of domain specific knowledge. *American Association for the Advancement of Science*, Washington, DC. http://public.sdsu.edu/CRMSE/Fisher_ aaas2000.html

ISO/IEC 14977:1996 (E). *Information Technology – Synctatic Metalanguage*. International Organization for Standardization, Geneva.

Jonassen, D., Mayes, J.T. & McAleese, R. (1993) A Manifesto for a Constructivist Approach to Technology in Higher Education. In T. Duffy, D. Jonassen, & J. Lowyck (Eds), *Designing Constructivist Learning Environments*. Heidelberg: Springer-Verlag.

Jonassen, D. H. (1996). *Computers in the Classroom: Mindtools for Critical Thinking*. New Jersey: Prentice-Hall.

Li X., S. Szpakowics, S. Matwin, A WordNet-based Algorithm for Word Sense Disambiguation. *In Proceedings of IJCAI-95*. Montreal, Canada, 1995.

McAleese, R., Grabinger, S. & Fisher, K. (1999). The Knowledge Arena: A Learning Environment that Underpins Concept Mapping. *American Educational Research Association*, Montreal, Canada.

Multibook (2004). *The Multibook Project.* http://www.multibook.de

Nastase, V., S. Szpakowics, Word Sense Disambiguation in Roget's Thesaurus Using WordNet. *In Proceedings of the NAACL WordNet and Other Lexical Resources Workshop*. Pittsburgh, June 2001.

Novak, J. D. (1998). *Learning, creating, and using knowledge: Concept Maps as Facilitative Tools in Schools and Corporations.* Mahweh, NJ: Lawrence Erlbaum Associates.

Novak, J. D., & Gowin, D. B. (1984). *Learning How to Learn*. Ithaca, New York: Cornell University Press.

Peña, A. O., Rubio, A. M., & Sanchez, A. L. (1996). *Los Mapas Conceptuales en el Aula*. Buenos Aires: Magisterio del Río de la Plata.

Rocha, F. E. L., Costa Jr, J. V., & Favero, E. L. (2004). A New Approach to Meaningful Learning Assessment Using Concept Maps: Ontologies and Genetic Algorithms. *In Proceedings of First International Conference on Concept Mapping*, Navarra, Spain.(2004).

Rocha, F. E. L. & Favero, E. L. (2004). *CMTool: A Supporting Tool for Conceptual Map Analysis*. In Proceedings of World Congress on Engineering and Technology Education 2004. Santos, Brazil, March, 2004.

Yudelson, M. V., Yen, I., Panteleev, E., Khan, L. (2003). A Framework for an Intelligent On-Line Education System. *In Proceedings of the 2003 American Society for Engineering Education Annual Conference.*