Entity-Relationship Model and Hypermedia Concept Mappings

Marcelo Zanconi Mercedes Vitturini *
Departamento de Ciencias de la Computación
Universidad Nacional del Sur
Av. Alem 1253
Bahía Blanca
Argentina
{cczanc,mvitturi}criba.edu.ar

Abstract

Modeling the thought has been subject of deep research in many different, and apparently disimile areas, such as psychology, history, computer science and antropology. By the middle of the century, machines were developed in the aim of modeling this thought and capable of answering questions in a "human way".

This universal aim seemed to be too huge and models were designed to answer very much reduced problems. We can find in the area of artificial intelligence different tools which can model some problems. In particular, hypermedia concept mapping serves as a tool to show a concept structure of a subject and the relation among other more basic concepts.

In the area of database the entity-relationship model has overcome a universal tool to deal with information processing problems. Completely far away, the area of teaching and learning has developed many tools to afford the problem of modeling the thought by understanding the way we think and we layout ideas.

In this work, we analyse this two models, compare them and arrive to some conclusions of their similarities and differences; even if their application areas are absolutely different, they share a commun structural point and we also analyse the usefulness of solving a problem using one model by knowing the other.

Keywords: entity-relationship model, concept mapping, data abstractions.

*Both members of the Education in Informatics Research Group from Universidad Nacional del Sur

Entity-Relationship Model and Concept Mappings

1 Motivation

The aim of this work is to establish an analogy or correspondence between two information models: hypermedia concept mapping, (HCM), [3] and [4], and entity relationship model, (E/R), [2], [?]. Besides this, as we enforce the benefits of working with modeling and abstrations since first courses, we are sure that by knowing one of the models it is easy to develop the other.

The concept of classifing and grouping objects is, certainly, not new. In the area of computer science, the task of grouping grew up since computers were developed as a good repository of data where the availability of data was important but also the need of rearranging data according to different requirements.

The first of this models is perhaps the hierarchical model where data were arranged in trees or hierarchies establishing some kind of dependency between father and son, that is, where information could be gathered through tree traversal from parents to sons.

This model was extended to more rich structures such as graphs and by the middle of the sixties the network model was developed. In this model data is related to another data through some kind of link, but there is no hierachy and data can be gathered by graph traversal no matter the sense.

By the beginning of the century, the E/R model was born, [2]. In this model, we can distinguish independent data, which represent object in real life, and linking data, which represent weak data to serve as a connection among independent data.

In any of these models the idea of grouping similar data and specially linking data is of most importance and is one of the greatest difficulties observed in data base curses. A good model is the key for any information system, as it guarantees that correct and consistent information would be extracted from the stored data.

Much more modern is concept mapping, CM, [5], specially developed as a tool for showing the different concepts related to a subject. This tool is nowadays extensively used in high school. In practice it may be useful in classroom as a mean of summarizing or showing a panoramic view of a subject or even as an examination tool. These mappings were extended by Seas, [3], with the notion of views of maps, -more general or specific maps-, and the use of hypermedial techniques which turn them into a much more rich expressive tool, [4], creating the HCM

Nowdays, people studying about education think that the key in the learning activity is “learning to learn”, “learning to think” and “understanding the meaning”. In that definition of learning, they refer to “significant or cognitive” structures to define the knowledge a person has on a certain subject, [6] and [5].

Using CM while learning follows these ideas:

1. acquire the ability of identifying the main concepts of topic under study,
2. identify the links among concepts, that is to say “modeling” in a graphical manner the structure of the concepts one has in mind.

Our final idea is that by knowing how to draw a HCM one can build better E/R models. This may be very useful for database courses.

2 Concept Mapping

CMs were developed by Novak in [5] and might be used as a tool for modeling thoughts and laying out the teaching and learning process. They present knowledge by meaningful relationships among concepts, namely *propositions*.

According to Ausubel, [1], knowledge is the ability to understand the meaning of something by capturing the cognitive structure of the problem. These cognitive structures are used to designate knowledge, its organization and its conection to other knowledge. Ausubel recalls that new ideas can only be captured if there are previous concepts from which one can build or understand or establish a relationship to these new ideas. It is important that basic knowledge form the platform of a *knowledge pyramid*, on the bottom of this pyramid one can find the most specific or complicated knowledge, while on top one can find more general and abstract concepts. It is interesting to note that some *revolutionary knowledge*, (such as some surprising results in physics or astronomy), have departured from a whole new theory, and perhaps this explains why people resists to believe it.

The cognitive structure of a problem is generally displayed as a graph or an inverted tree , (or more generally as trees are drawn in the area of computer science), with its root on the top, representing the most general concept and leaves the most specific. In the middle one displays related concepts or propositions.

A proposition consists of two or more concepts related by a word creating a semantic unit of knowledge. This unit is much more than the sum of the individual concepts involved, such as is expressed in Gestalt Theory. Certainly, one can only build new units from very well known previous concepts. Ausubel distinguishes *unitary concepts* and *composed concepts*. The former are isolated concepts representing objects, events, or real facts which are designated by symbols while the latter are much more complex since they involve various concepts and a relating structure. This classification is by no means strict, since one concept may be unitary for a problem while composed to another, depending on what is intented to be expressed or what is suppossed to be known.

A problem is then reduced to a small set of important and interrelated ideas where one must focus attention.

From a syntactic point of view, a CM is a graph $G = (V, N)$, where N is the set of nodes and V is the set of arcs linking the nodes. From a semantic point, each arc is labelled by a proposition word and reflects a hierarchy or an inclusive order from top to bottom; but unlike trees some nodes may have more than one “father”.

CMs are a text tool which can be used to summarize a particular subject. In this sense, it has been shown to be a good tool in the teaching-learning process. Señas et al. [4] have shown a wider class of CM: *hypermedia concept mappings*, which are CMs enriched by all means of information: textual, visual and audio, where some nodes in N may itself be another CM, constituting the concept of *view* and defining *internal arcs* and *external arcs*. The former serve as a mean of connecting concepts within a map while the latter connect concepts among different maps.

As seen, there are two different classes of elements: concepts and links.

2.1 Concepts

Concepts are described as a regularities which show the objects of the problem. As stated by Novak, one of the most relevant keys in knowledge acquisition is human perception to abstract information and represent it by symbols showing the regularities of facts and objects.

Concepts exist by themselves and have their own meaning, presenting ideas which are understood aside the problem. Some concepts are more general than others; by comparing concepts one can realize that some have more detailed information and may include them in a bigger class. This process may go further, resulting in a hierarchy of concepts.

2.2 Links

Links or propositions are joining words which show the relationships among concepts. Links together with the concepts involved build a semantic unit which in fact is the smallest semantic unit with a value of truth.

Links go from a concept to another, joining a concept with a more inclusive one. Even when, by definition, we do not draw a directed arch, we must read it as going from top (more general) to bottom (more inclusive).

Joining words label the archs among concepts. These words establish the main relationship among them and must be strong enough in order to create the idea pretended to be shown.

2.3 Creating a Hypermedia Concept Mapping

When building HCMs one must first identify concepts. Then, one must put them in certain order, according to an inclusion relationship. This relationship is by no means absolute and depends on the point of view of the problem being described. We can generally find a *root concept* which starts it all. In this first step, the key idea is to display all concepts describing the problem.

Once concepts have been ordered, one must set up relationships among them, by finding linking words. These words, such as "is formed of", "contains", "is a", show the inclusion relationships between concepts. Generally, these words act as propositions in the mathematical sense.

In HCM, concepts are organized in subsets called *views*. This partition is obtained from a semantic analysis of the concept set, the hierarchy of concepts and the cardinality of each of the classes, [4].

Concepts in a HCM might be associated with different "appearances", such as a photo or a brief explanation illustrating the concept. Novak's concepts are merely described through its relationships among other concepts.

In this way, a hierarchy is built based on the order of inclusion. Novak's CMs show each concept as a unit while HCM might override the hierarchy since a concept might have its own CM describing it. In [4] a methodology is proposed to construct HCM.

3 The Entity-Relationship Model

The entity-relationship model, (E/R) is largely used in the world of database systems as a designing tool. It primary focuses as a "data model", providing a notation for data description and a clear way of displaying relationships among them. The idea in the construction of an E/R diagram is to transform the part of real world pretended to be modeled into a collection of special objects: entities and relationships.

The object of this work is not to explain the E/R model, but to see its similarity to conceptual mappings (CM). For this, we first shortly explain each of its components.

3.1 Entities and Entity sets

An *entity* is an object that exists and is distinguishable from others. A group consisting of similar entities forms an *entity set*. Each entity set is characterized by a set of properties, called *attributes*. For each entity in the set and for each attribute one can associate a value from the domain of values for that atribute. Thus, each entity can be recognized by this set of values for attributes. An atribute or set of atributes whose values uniquely identify each entity in an entity set is called the key for that entity set.

3.2 Relationships and Relationship Sets

Two or more entity sets can be linked to create a bidirectional connection among them. This connection is called *relationship*. A relationship is an ordered list of the entity sets involved. Formally, we say that if it exists a relationship R among entity sets $E_1, E_2, \ldots, E_k$, then an instance of R is a set of k-tuplas $(e_1, e_2, \ldots, e_k)$, where e_1 is in set E_1, e_2 is in set E_2 and so on; The value k represents the *cardinality* of the relationship, i.e. the number of entity-sets involved, the most common case being $k = 2$,

Having a relationship among two entity sets E_1 and E_2, it is necessary to specify the *funcionality* or *multiplicity* of the relationship according to how many entities from E_1 can be associated with how many entities of E_2. The simplest and rarest form of a relationship is *one-to-one*. This means that for each entity in E_1 there is at most one associated entity in E_2 and vice-versa. Another type of relationship is *many-to-one*, i.e. one entity in E_1 is associated with zero or more entities in E_2, but each entity in E_2 is associated with at most one entity in E_1. Finally, the last type of relationship is *many-to-many*. Here there are no restrictions on the sets of entities that may appear in the relationship. We can extend these definitions to relationships R among $k, k > 2$, entity sets:

$$R : E_1 \times E_2 \times \ldots E_{i-1} \times E_{i+1} \ldots \times E_n \rightarrow E_i$$

by taking the $i - 1$ entity sets against E_i and thinking how is the funcionality.

Relationships may have their own attributes since there are properties which *belong to the relation and not to any of the entity-sets envolved*. From the model point of view, we can see the collection of attributes on a relationship as a new entity-set whose attributes are those ascribed to the relationship.

3.3 Creating an E/R model

Generally speaking, one can create an E/R model by defining the entity-sets, the attributes in each entity-set and the relationships among them. To do so, given a problem one must distinguish the information which has certain independence and reflects objects which exist by themselves, from information which depend on these entity-sets. By grouping objects of the same class and naming this class, one can create entity-sets. As each entity in an entity-set must be distinguishable, there must exist a subset of the attributes which can serve as a *key* for identifying it.

To create relationships, data links among entity-sets must be analyzed. These relationships are strongly dependent on the entity-sets and on the problem being modeled. Once relationships have been established, one must focus the problem of functionality, i.e., how many entities are in relation to other entities in the rest of entity-sets, [7].

The E/R model is a general designing tool which has its graphical representation. Each entity-set can be pictured as a square with its name inside, each attribute is pictured as a ellipsis with its name inside and connected to the entity-set. Finally, relationships are represented by diamonds with their name inside and connected to the entity-sets involved by an arc. Each arc can be arrowed or not depending on the funcionality of the relationship.

One of the hardest problems frequently observed in database courses is the subtle difference between entity-sets and relationship sets. It is clear that sometimes it is very easy to distinguish one from the other, but there are some cases where this difference is by no means clear. Entity-sets are primarily used to represent real-life objects or should reflect reality, such as books, students, cars, enterprises, while relationships are used to express linkage among entity-sets. In some sense, relationships are subordinated to entity-sets, since the former exist only if the latter exist. Sometimes the E/R model is expressed by *strong* and *weak* elements, but Chen's model includes two components as presented here.

There are some extensions to this model, including *agregates*, which are abstractions of higher level including relationships and entity-sets into a new object, which can be related to other entity-sets, [8].

4 Comparing the Models

Both, E/R model and HCM have the benefit of all graphic models: *a picture is worth a thousand words.* Given a set of conventions and well defined rules, it is clear to understand the problem and interchange knowledge between people involved in solving it.

Let us illustrate the steps involved in the construction of both models:

Entity-relationship diagram

1. Find the entity sets and their attributes.
2. Establish the relationships among entity-sets, their functionality and attributes (if any).
3. Find for each entity set and relationship set the key attributes.
4. Extend the model to present to the aggregates, if appropriate.

Hypermedia Concept Mapping

Basically, the needed steps to build a HCM can be found in [3], we summarize them here for clarity and completeness:

1. Identify the concepts.
2. Sort concepts by their inclusion relation and create the views.
3. Set the propositions linking concepts.
4. If wished, associate a hypermedia appearance to some concepts.

The first steps of both models are similar in escence. Both require abstracting the most relevant information in the problem. This process is by no way trivial and it needs a deep comprehension of the problem and a great experience in building these diagrams.

The second step in building an E/R and the third in HCM are similar too: establish the relationships among the main components.

The rest of the steps are special for the construction of each model. As a last step both models require a complete revision of the components and of any decision taken so far. Even when this step is not likely to be done, it is as important as revising the design of an algorithm: it guarantees fair results.

As can be seen, given a problem, both models pretend to extract the most relevant information and to stablish the links. In CM or HCM *generalization* is almost always present conducting to hierarchical models, while in E/R model information has a more flat structure, that is to say all entity-sets are at the same level, while relationships can be considered as a special class of entity-sets, dependent on them but not in a lower level.

Of course both models are semantically different. The hierarchy presented in CM or in HCM serve as a mean of distinguishing different levels of knowledge; by following the different levels of the map we recover inclusive information from the most general to the most specific. In HCM one can find another hierarchy level between two maps, linking much more complex concepts mentioned in a map, much widely explained in another map. This behaviour does not exist in E/R diagrams, which by definition, have a unique level.

We believe that if CM techniques are known by students it will much easier for them to build E/R model because both models pretend to organize information and ideas by relating concepts. These relationships act as cognitive bridges and express the underlying structure of knowledge. The process of recognizing objects and grouping them into a new more general and inclusive class is essential to both models, so this ability must be very useful if one knows how to build CM and the problem of building a E/R is presented.

However, we must recognize that CM is much more general than E/R model and nowadays is widely used in high schools. Our intention is to provide the E/R model as a special kind of CM devoted to information technology and database systems. As an exercise, we have experimented both techniques in class and have experienced that once students have captured CM modeling and its *know how*, it is much easier to develop information models.

5 Conclusion

From a structural point of view, both models share a common characteristic which is *data abstraction*. It is sure that if one is able to model a problem using the HCM model, one can say that has reached the ability of abstracting data and designing E/R diagrams. There are a number of common techniques present in the two models: abstracting, grouping, classifying, linking and naming.

From an applicative point of view, being a very general model, the HCM can be used as a learning tool in very different areas of knowledge, while the E/R model is circumscribed to data models or problems dealing with data managing.

The E/R model is quite useful as a tool to deal with problems based on data managing; on the other hand, the HCM can deal with graphic, audio, movies and other multimedia information. Nowadays the E/R model is used in the area of database design, -mainly at the university-, while the other is used as an educational tool, -to show the understanding of a problem or to summarize a subject, at school or higher school.

Even when both models do not have a formal definition, they are supported by the concept of a net, a special case of a graph, and data abstraction. We believe that HCM is not only a tool used in first courses, but also achieves some expected results from an educational point of view, effectively demonstrated by many experts in the area, [5] and [6], and trains students in abstraction techniques and viewing the different steps in problem solving, (what we call *target gapping*).

All these techniques are undoubtly useful in any application area but we are particularly interested in database system area. Data abstraction is very hard to achieve, but CM are widely used and may be applied to different subjects, since the begining of high school.

Each CM may be analysed according to the madurity degree of the student, planning more complex maps as the student gets more familiar to them and also achieving a more specific map as knowledge is deeper.

References

[1] D. P. Ausubel, J. D. Novak, H. Hanesian. *Psicologa Educativa*. Mxico, Trillas. 1989.

[2] Peter P. Chen. *The Entity-Relationship Model: Toward a unified view of data*. ACM Transactions on Database Systems. Vol 1. N. 1, pp 9-36. 1966.

[3] Perla Señas, Norma Moroni, Mercedes Vitturini, Marcelo Zanconi. *Hypermedia Concept Mapping: A Development methodology*. ICTE "96. New Orleans. United States of America.

[4] Perla Señas, Norma Moroni, Mercedes Vitturini, Marcelo Zanconi. *Combining Concept Mapping and Hypermedia*. ED-Media "96. Boston. United States of America.

[5] Joseph Novak, D. Bob Gowin. *Aprendiendo a Aprender*. Ed. Martinez Roca. 1988.

[6] Antonio Ontoria. *Mapas Conceptuales. Una tcnica para aprender*. Narcea S.A. Madrid. 1994.

[7] Jeffrey Ullman. *Principles of Database and Knowledge-base systems*. Computer Science Press. New York, 1989.

[8] . J. M. Smith, D.C.P. Smith. *Database Abstractions: Aggregation and Generalization.* ACM Transactions on Database Systems. Vol 2. N. 2. Marzo 1977.