Knowledge Discovery in Multimedia Design Case Bases

Mary Lou Maher and Simeon Simoff
Key Centre of Design Computing
Department of Architectural and Design Science
University of Sydney NSW 2006
fax: 61 2 9351 3031
{mary,simeon}@arch.usyd.edu.au

ABSTRACT: Case-based reasoning is a problem solving paradigm based on the concept of analogy with application to a memory of specific problem solving experiences. Recent research in case-based design has lead to formalisms for representing design cases, indexing schemes and adaptation knowledge, however, each application requires that the relevant representation be hand crafted and manually entered. The development of the indexing scheme and the adaptation knowledge can be automated using a knowledge discovery technique. Knowledge discovery techniques can be applied to existing design case bases to generate indices, ontologies, and rules as needed by the case-based reasoner. Such techniques are presented in the context of the SAM project, a multimedia case base of building designs.

Keywords: case-based reasoning, multimedia, knowledge discovery, design

1. Introduction

Case-based reasoning is an approach to problem solving that is based on the concept of analogy, where problems or experiences outside the one we are currently dealing with may provide some insight or assistance. Through analogy, we may be reminded of a window design when designing a door to a balcony with a view. Analogy is a way of recognising something that has not been encountered before by associating it with something that has. Considering analogy from the perspective of memory and reminding has led to the concept of memory organisation as a guideline for computer representations. The study of memory organisation, its use as a basis for new problem solving and generating explanations, has led to models for representing experience in computers. This area of AI is called *case-based reasoning* [1]. Case-based reasoning is a research area within AI and has been applied and used in real-world situations.

An overall model of case-based reasoning, as shown in Figure 1, illustrates the role of design cases in the case-based reasoning process. Representing design cases requires an abstraction of the experience into a symbolic form that can be manipulated by the reasoner (either computer or human). Various abstractions have been developed based on models, methods, and philosophical approaches to understanding design, designers, and design experiences.

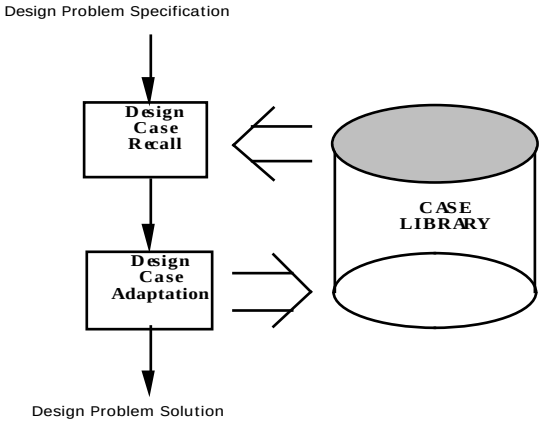

Figure 1. The overall process model of case-based design

The process of recall requires an indexing scheme for retrieving cases from the case library. An efficient indexing scheme is not important for a small library, where the common approach is to keep a list of pointers to each case. In a large case library the indexing scheme is critical and a common approach is to develop a network or tree

representation of case features that serve to partition case memory. The indexing scheme may be generated through knowledge acquisition techniques of interviewing the expert to identify the critical features [2], or machine learning techniques to identify the most discriminating features by induction. We propose a knowledge discovery approach in which the cases can be described in a variety of ways including text and images.

The process of adaptation requires knowledge about the domain for modifying and evaluating the retrieved case to be feasible in the new design context. Adaptation knowledge can be in the form of heuristics, causal constraints, or object-oriented models. The knowledge is in a form that can be applied to the design case representation. The approach to developing the adaptation knowledge is similar to the indexing scheme: the expert provides the rules and/or models of relevance. We propose that the knowledge discovery approach can be used to find hueristics for case adaptation.

2. SAM: a multimedia case library

The conceptual design of structural systems for buildings is still largely unsupported by computer-aided design. Early efforts in applying AI techniques to structural design, eg HI-RISE [3], identified AI as an alternative approach to computer support for design to the more conventional finite element analysis or optimisation techniques. However, the use of AI techniques in the professional practice of conceptual structural design in not widespread. This does not necessarily mean that AI does not hold promise as a conceptual design aid, but that there needs to be more focus on the nature of the computer support and the amount of effort needed in developing and maintaining the support systems.

Case-based reasoning as a support environment for conceptual design of structural systems is attractive for two reasons:

- the knowledge is represented as design cases that can be proprietary and/or familiar to the engineering consultant, and
- the knowledge as case memory can be maintained and updated automatically with the use of the system.

The application of case-based reasoning to structural design, eg CASECAD and CADSYN [4], has shown that the development of these case-based reasoning systems has to take into consideration the representation of design cases and the representation of the knowledge needed to recall and adapt the design. These systems were developed through extensive contact with the expert structural engineers involved on the projects in the case library and through the application of previous research in memory organisation and knowledge-based design. The resulting systems contributed to an understanding of case-based design but did not provide libraries or systems useful to design engineers.

SAM is a project in which a design case library is being developed on the World Wide Web for use by students. As a learning environment the emphasis is on effective use of multimedia information, navigation and interactivity for case browsing and retrieval, and ease of adaptation by the students. SAM currently has 25-30 cases that present the structural design of high rise and wide span buildings in Australia. The case is organised according to structural design principles and specific structural design systems, as illustrated in Figure 2. The case is presented as a combination of text, tabular data, and images of photographs, sketches, and CAD drawings.

A building design case is a complex system. In SAM a case is decomposed into parts: general project informaton, functional systems information, and structural system types information. The general project information includes:

- a summary of project data,
- the design requirements,
- alternative designs considered, and
- major design decisions/justifications.

The major functional systems included in a SAM case are: the lateral-load resisting system, the gravity load resisiting system, and the foundation system. Each functional system is further described as having a primary and secondary system. In addition to organising information according to function, each design case has information about the major structural system types used in the building, eg. trusses, arches, butresses, etc. An example of a description of a structural system type is shown in Figure 3.

3. Knowledge Discovery as Part of Knowledge Engineering

Knowledge discovery (KD) is the process of examining a data source for information that one is unaware of prior to the discovery. This spans the entire spectrum from discovering information of which one has no knowledge to where one merely confirms a well known fact. KD involves the identification of potentially useful and understandable patterns in this data [5,6].

Figure 2. Front page for the ANA Hotel case

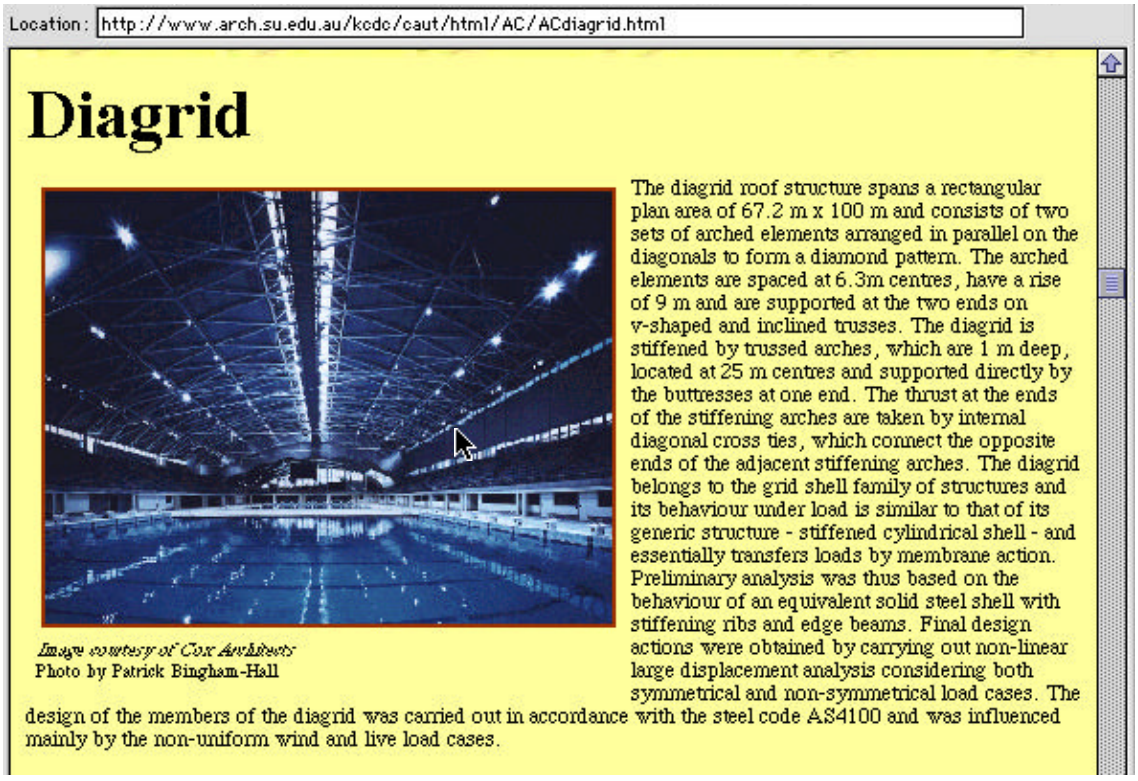

Figure 3. Description and illustration of a structural system in SAM

KD is similar to machine learning [7] with the subtle difference being that the data used as input to a machine learning program is carefully prepared in format and content and the input to a knowledge discovery program is data that was stored for purposes other than to support machine learning. Thus, we can say that *knowledge discovery is a machine learning where the training set is replaced by a database, knowledge base or case library*. This idea with respect to the case library is illustrated in Figure 4.

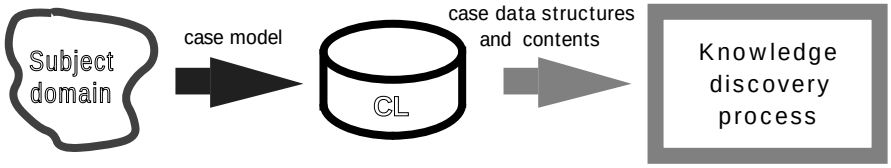

Figure 4. General diagram of knowledge discovery in case bases

The KD process in case libraries is organised into the following basic steps, as shown in Figure 5.

- *Case selection*. A case library contains a variety of cases and each case contains a variety of data not all of which is necessary to achieve each knowledge discovery goal. Thus the first step is to index the cases and select the data that will be processed by the KD techniques. The selected data types may be organised along multiple cases, which requires the creation of a separate transition cases.
- *Case transformation*. Once the case is identified, it is necessary to perform certain transformations, such as removing duplicate records, organising data in desired way, conversions of one type of data to another, to definition of new case attributes. These attributes are derived by applying either mathematical or logical operators on the values of one or more case attributes.
- *Case examination and mining*. The transformed cases are subsequently examined using one or more KD techniques in order to try extracting the desired type of information. The way the examination is performed depends on the goals of the case mining.
- *Result interpretation and discovery absorption*. The extracted information must finally be analysed with respect to initial goals and tasks. For example, if a classification criterion has been developed, during the result interpretation step the robustness of the criterion is tested using one of the established error estimation methods. During this step it is also determined how to best present and incorporate the evaluated results. For example, if the case reasoning engine is built as a set of predicates then it is suitable to express the results also in predicate form and add them to the case reasoner.

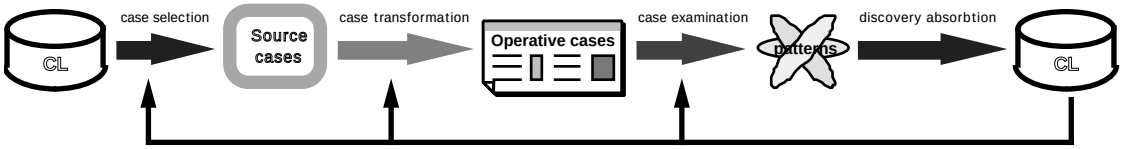

Figure 5. The process of knowledge discovery in case libraries

Following Williams and Huang [5], we identify the following key components of the KD process:

- the *case collection* which includes the initial case library, the source case set, derived from the case library by particular indexing and retrieval of relevant cases, and the operative cases. The case collection can contain an important implicit knowledge, such as the domains of textual attributes and the value ranges of numerical attributes, distributions of attribute values, and relations between attributes.
- the *knowledge representation scheme* which provides the framework for a more structured representation. The selected scheme is related to the case representation schemes used in the case collection. A multilayered case representation is a combination of connected representation schemes for each layer.
- the *knowledge discovery function* which determines the features and groups of features that are of interest. Discovery functions vary from a very simple threshold discriminating function which separates the range of an attribute into two distinct qualitative regions to a sophisticated fuzzy relation which maps a range of numerical values onto a set of words, thus converting a numerical case attribute into a more generic linguistic attribute.
- the *set of operations* which manipulate and transform case data. It includes case indexing and retrieving algorithms used for forming the source case set and the operative cases.
- the *set of goals* which influences the types of operations and discovery functions that are used in the knowledge discovery process.

The basis of a KD technique is the discovery method, which computes and evaluates groupings, patterns and relationships in the considered case data set. There are numerous discovery methodologies and discovery algorithms,

based on procedures drawn from inductive logic, statistics, cluster and discriminant analysis, machine learning. We classify them with respect to the *goals* they have to achieve and the *knowledge* they produce.

Depending on the goals of the KD process we distinguish the following two categories:

- **verification-driven search** where the goal of the KD process is to verify particular facts about the cases included in the library. For example, let us suppose that we need to design a railway bridge. Considering a case library of bridges we have to verify that the cases there are related to the railway bridge construction. Thus the verification-driven case mining has an analogy with the supervised learning. Enabling techniques are based on deductive reasoning and on a variety of statistical inference methods, including cluster analysis and discriminant analysis.
- **expectation-driven exploration** where we vaguely know what we are looking for, ie. we formulate an hypothesis, which is either refined during the exploration, partially or completely reformulated or finally rejected. The strategy is similar to the iterative search in machine learning, except that the quality function can be changed dynamically.

In the context of the knowledge the KD process produces, we distinguish the following categories:

- **discovering regularities -** this group of inductive methods find qualitative and quantitative patterns among sets of parameters drawn from design case descriptions. The techniques applied can be based on analog clustering, using the self-organising maps from neural-network domain, or relational analysis, based on binary clustering, sequential pattern extraction, based on time-series analysis. Discovered regularities can be used for predictive modeling of missing case parameters.
- **discovering structures** - this group of methods is related to the structural data in design cases. The idea is to identify concepts describing interesting and repetitive substructures within structural design cases. The algorithms use graph-oriented data representation and graph-based induction.
- **discovering functions -** this approach can be implemented via different schemes, including symbolic reasoning algorithms and neural networks. For example, while analysing the structural load attribute in the cases in the case library, the system may discover a functional dependence which can be used to discriminate or correct new cases. However, there is a difference between discovering functions and fitting functions. Standard regression analysis start out with a function and minimise the square of the error between the data points and the curve. The function discovery technique works without assuming a particular shape of the function.
- **discovering rules** - these methods are oriented on generating *association rules*, where the rule antecedent includes a judgement considering a set of case attributes and the consequent includes a judgement about a single case attribute. The rules might be exact or with some degree of uncertainty. They may be used further for guiding the discovery process. The algorithms are based on iterative induction. The whole process may become recursive, ie. the process may continue with finding new rules from the sets of discovered association rules.
- **deviation detection** - this technique extracts the exemptions, ie. identifies why some cases cannot be put into specific category. The methods are based on statistical analysis.

Many times different KD techniques can be used cooperatively. Generally, the more algorithms are employed, the higher the likelihood of effective and accurate results. Regardless of the algorithm, the approach to knowledge discovery depends on the type of design case representation and the size of the case library, and the specifics of particular design domain.

Virtually all techniques for knowledge discovery in text data are based on the assumption that relevant concepts co-occured in the same text. This co-occurence of word types in text is justified by estimating different similarity statistics. Similarity coefficients are often obtained between pairs of distinct terms based on coincidences in term assignments to examined texts. When pairwise similarities are obtained between all term pairs the terms with sufficiently large pairwise similarities are grouped into common classes.

Thus a common approach for discovering terminological patterns from documents is to parse the document text, identify the content-carrying terms, cluster the inter-term dependencies, and build a conceptual hierarchy. This approach is used by Dong and Agogino [8] to learn a conceptual hierarchy of concepts in mechanical engineering design from design documents. The process involves an unsupervised learning stage in which related terms are classified into conceptual cells. The cells are based on terms that appear together in the same document, assuming that they connote similar meaning. The terms in the conceptual cells are then used to construct belief networks using a heuristic initial network - constituting a supervised learning approach.

4. Knowledge Discovery in SAM

Effective knowledge discovery from design cases requires more detailed classification of data types to be analysed. We describe briefly each particular type of case data that has to be investigated.

- *Multimedia design data* includes graphics, image, speech, music, animation and other forms of audio/video information. Some of the graphics data is represented in the form of vector and spline approximations, which incorporates geometrical knowledge about the objects in the image.
- *Long text data* includes any textual description of the design, including design notes, books, articles, standardisation documents, comments, e-mail communications, or other kinds of documents.
- *Structure-valued data* includes attribute-value sets (table descriptions), list-valued data, eg. the list with the names of files related to the case, and nested structures (hyper links data).

The case library in SAM incorporate structure-valued data, long text and multimedia data. The structure-valued portion of the case data is the easiest for data mining. The beginning of every case includes lists attribute-value pairs. This data could be processed with the data mining techniques applied to database tables. The long text data type is the basic data type in SAM cases. The above discussed approach could be adapted for "knowledge discovery - case updating" cycle, which is illustrated in Figure 6.

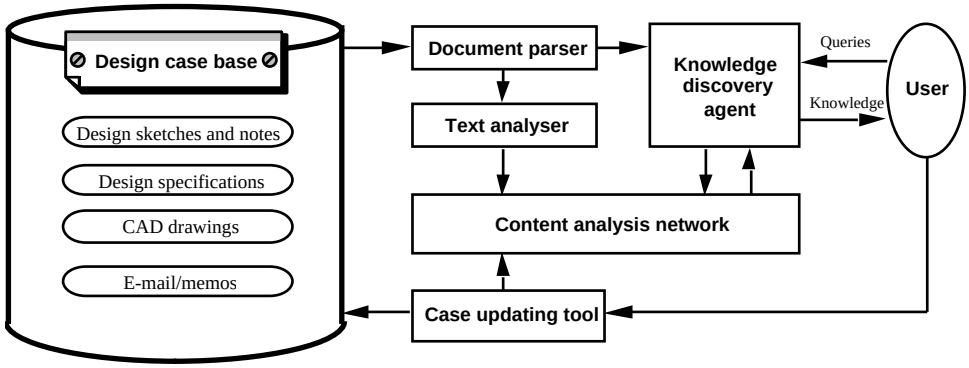

Figure 6. Text-analysis for constructing design case representations (adapted from[8])

There are several ways to discover knowledge patterns and extract essential features from multimedia data. For an image, the size, colour, number of objects, the shape of the objects and the corresponding labels can be extracted, though with errors by some pattern recognition algorithms, spline approximation algorithms, or object aggregation and decomposition techniques. For an audio segment, regularities can be summarised based on the approximate patterns that repeatedly occur in the segment. However, it is a challenging task to analyse and to extract the implicit knowledge stored in the multimedia data.

The application of knowledge discovery techniques will extend the efficiency of the retrieval process and allow the indexing scheme to be dynamically updated when new cases are added to the library. In addition, the discovery of rules from SAM's library will facilitate the evaluation of adapted designs. The knowledge to be discovered includes:

- thesaurus whose terms can serve as indices to SAM's cases
- semantic objects (words, word combinations) that can provide the basis for the development of design case description language.
- semantic-based concept query interface for browsing SAM library.
- ontologies for structural design that can serve to partition SAM's library for efficient retrieval.
- rules that can serve to evaluate the feasibility of an adapted case.

References

[1]. Koloona, J. L.: "Case-Based Reasoning", Morgan Kaufmann, New York, 1993.

[2]. Jackson, P.: "Introduction to Expert Systems" (2nd ed.), Addison Wesley, Reading, MA, 1990.

[3]. Maher, M. L.: "HI-RISE and Beyond: Directions for Expert Systems in Design", CAD Journal, Vol 17, No. 9, 1985.

[4]. Maher, M.L., Balachandran, B., Zhang, D.M.: "Case-Based Reasoning in Design", Lawrence Erlbaum Associates, New Jersey, 1995.

[5]. Williams, G. and Huang, Z.: "A Case Study in Knowledge Acquisition for Insurance Risk Assessment using a KDD Methodology", Data Mining Portfolio - TR DM 96023, CSIRO, 1996.

[6]. Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P. and Uthurusamy, R. (eds): "Advances in Knowledge Discovery and Data Mining", AAAI Press, Boston, MA, 1996.

[7]. Holland, J.H., Holyoak, K.J., Nisbett, R.E. and Thagard, P.R.: "Induction: Processes of Inference, Learning and Discovery", Computational Models of Cognition and Perception., MIT Press, Cambridge, 1986.

[8]. Dong, A. and Agogino, A.: "Text Analysis for Constructing Design Representations", in J Gero and F Sudweeks (eds) Artificial Intelligence in Design '96, Kluwer Academic, Holland, pp 21-38, 1996.