Concept clustering for cooperation

U.Srinivasan, A.H.H.Ngu, T.D.Gedeon
School of Computer Science & Engineering
University of New South Wales
P.O.Box 1, Kensington 2033, NSW, Australia.
e-mail: {uma, anne, tamas}@cse.unsw.edu.au
Phone: (612) 385 3970 Fax: (612) 313 7916

Abstract

Most heterogeneous Clinical Information Systems share a strong semantic similarity in spite of their autonomy and heterogeneity. To exploit this semantic similarity, we propose a concept discovery approach based on statistical clustering techniques to develop a generic conceptual schema to establish interoperability. The usage pattern of users is not just statistical information, but carries contextual information that can be exploited during the concept discovery process.

Keywords

Cooperative information systems, schema integration, semantic similarity, concept discovery, concept clustering, interoperability,

1 INTRODUCTION

Different organizations within a large enterprise usually use substantively different, incompatible information systems. The differences (of structure and content) are due primarily to the different organizational origins of the various information systems. (For example, the IBM Corporation uses nearly 200 different information systems for its various computer businesses alone! (Bhaskar, 1995). In spite of these substantive differences there are actually strong semantic similarities in the *information* in the information system. In a large enterprise such as a hospital, individual clinical departments often manage their own information systems which typically cannot be disturbed and form what are referred to as legacy information systems (Brodie, 1992). These information systems are often structurally and semantically different. Consequently, information sharing across these information systems (or databases) is

hindered by their heterogeneity and autonomy. In spite of their heterogeneity, these systems share a strong semantic similarity due to two factors. Firstly, they all store and manage the same type of data, viz., patient data and secondly, the basic specification for each individual system is initially put together by users such as doctors who are equipped with similar background knowledge. With the increase in demand for interoperability among existing systems, a mechanism that can establish cooperation amongst the heterogeneous clinical databases becomes crucial. In this paper, we propose a concept discovery approach that exploits this semantic similarity and allows one information system to use information from another.

Heterogeneous database research has largely dealt with the difficult issues related to resolving structural and semantic conflicts (Kim, Seo, 1991) (Siegel, Madnick, 1991) (Sheth, Kashyap, 1992) and developing a global schema to provide integration (Collet, Huhns, Shen, 1991) (Ahmed et.al l991). More recently, the thrust has been in establishing intelligent cooperation among heterogeneous systems to achieve interoperability (Papazoglou, Laufman, Sellis, 1992) (Johannesson, Jamil, 1994). One of the means of achieving this is by negotiation between agents sharing information (Milliner, Papazoglou, 1994). Another suggested method is sharing semantic values as a unit of exchange that facilitates semantic interoperability (Sciore, Siegel, Rosenthal, 1994). Yet another possibility for establishing cooperation, one related to ours, is to develop a conceptual schema for cooperative information systems by using a semantic reconstruction approach. However, semantic reconstruction is not possible using the same rules that were used to develop the information systems in the first place, since the knowledge that was used originally during the initial specification and design phase is no longer available or valid as pointed out by Meersman (1994). (This is of course true in many domains). This is because over a long period of use, perhaps years, both the users and the underlying database structures change with the evolution of business rules and technology, and changes in user requirements.

Here we present an approach that uses the current structure of a part of CIS to somehow “discover” a set of concepts common to several CIS. This approach which we have labeled “conceptual clustering” uses an audit trail to keep track of usage patterns of a particular group of CIS. In this approach, the current structure is the kernel from which inductions are made about relevant new concepts. Existing database structures play an important role as they depict the domain knowledge of the user at the time of specifications. The attributes/properties of entity definitions and relationships among entities/objects in each CIS are valuable sources of information that reflect domain knowledge. Establishing an audit trail to gather usage data needed for concept clustering has the advantage that it can be technically specific and induced. Data pertaining to usage pattern is not just statistical information, but carries a cognitive content that can be exploited during the concept discovery process..

In the heterogeneous database context the search for common concepts poses two substantive problems: first, the usual problems of data semantics and domain mismatch (Siegel, Madnick, 1991) (Saltor, Castellanos,Garcia-Solaco. 1992); second, the confidentiality of patient clinical data pose difficult ethical and social issues. Our approach attempts to treat both these problems, with the confidentiality issue being treated as a hard constraint on any solution.

Concept discovery can be thought of as a form of inductive learning (Stepp III, Michalski, 1986), where learning takes place by a process of discovery that searches for similarity among database objects in potential interoperable databases. Although many researchers have proposed applications of machine learning to discover hidden relationships of data (Shapiro, Frawley, 1991) (Zytkow, Baker, 1990), and to learn new properties of objects (Quinla, 1990), till recently, not many attempts have been made to use these techniques in the area of interoperable databases. We use a knowledge discovery method that uses statistical clustering to bring out the similarity among the database objects that are clustered.

In order to establish a cluster of objects, each object is characterized by a set of variables. The variables are chosen such that they are good indicators of both the structure and usage of the database objects and also serve to establish similarity measures among the objects that are clustered. We then use a clustering algorithm that partitions these objects into clusters using the values of the variables (data gathered during the audit trail mentioned earlier, are the values of these variables). The clusters thus formed are meaningful in such a way that each cluster actually represents a generic concept in the application domain. The set of application specific generic concepts so discovered constitutes the reconstructed conceptual schema that can support interoperability among the information systems that were clustered. This generic conceptual schema can then serve as the layer that provides the required external view to the users of the cooperative information systems in the application domain.

The clustering algorithm proposed has been tested using data gathered by setting up audit trails over a number of existing CISs. In order to evaluate the quality of clusters produced by the algorithm, a simple informal basis is used at this stage. The informal basis for evaluation is to compare the results of the clustering algorithm with the generic concepts identified manually by a human expert. The simple comparison of clusters identified manually by the human expert and the clusters discovered automatically shows that the quality of concept clusters obtained automatically is found to be better across systems that manually interoperate than those clusters discovered across systems that do not manually interoperate.

Concept discovery in databases deals with issues relevant to several other fields including database management, machine learning, expert systems and statistical

analysis (Frawley, Shapiro, Matheus, 1991). Some background information on related areas of knowledge discovery and machine learning are described in the next section. Our approach to developing a generic conceptual schema is then outlined in section 3. This is followed in section 4 with a description of the actual clustering algorithms used to derive the generic concepts. Results of some experiments are also discussed in this section. Finally, section 5 presents the conclusions and future research directions.

The applicability of these generic concepts in providing customized external view to the users of interoperable systems are discussed in a working paper (Srinivassan, Ngu, 1994). Although all the examples in this paper are specific to the application domain of a hospital, the approach may be general enough to be applicable in any application domain where there is a Data Base Administrator (DBA) or a Data Manager with some knowledge of the data structures and usage patterns of users in the application domain.

2 RELATED WORK: KNOWLEDGE DISCOVERY

Discovery algorithms are procedures that extract interesting patterns in data and typically process raw data contained in a database. In practice, we have found that the discovery system must also use additional information about the form of data and the constraints on it. Because discovery is computationally expensive, additional knowledge about the form and content of data, domain(s) described by the database, the context of a particular episode and the purposes being served by discovery can be used to guide and constrain the search for interesting knowledge (Frawley, Shapiro, Matheus, 1991). Shapiro's book (Shapiro, Frawley, 1991) offers a good collection of different categories of discovery programs. However most of them use the basic data mining approach that only deals with raw data values, to discover knowledge in the form of laws, rules and structure. Moreover, most of these systems look for patterns that apply to individual instances. Other domain specific discovery methods (Gonzalez, 1991) (Siegel, Sciore, Salveter, 1991) work towards developing a knowledge base from existing computer resident data sources.

Creating a meaningful classification is a form of learning from observations and its goal is to structure the given observations into meaningful categories or clusters. Cluster analysis is a generic term for a set of techniques which produce classifications from initially unclassified data (Everitt, 1980). The reasons for classification may differ from user to user - a possible reason for the existence of a large variety of clustering algorithms. The central notion used for creating classes of objects (or clusters) is a numerical measure of similarity of objects. Clusters are collection of objects whose intra-class similarity is high and inter-class similarity is low. Traditional

clustering methods used numeric data without the use of background knowledge. Conceptual clustering (Michalski, Stepp, 1985) is an approach in which a configuration of objects forms a class only if it can be described by a concept from a pre-defined concept class. In an extension to this work (Stepp III, Michalski, 1986) it has been shown that that in order to create meaningful classifications, a system must be equipped with background knowledge, which includes goals of classification, classification evaluation criteria, deductive and inductive inference rules.

The concept discovery approach presented in this paper draws a number of ideas from Stepp and Michalski (Stepp III, Michalski, 1986). The main thrust of our approach is that the search for concepts is guided by information in existing CIS. As we are dealing with legacy systems, existing database structures and usage patterns provide sufficient background knowledge that can be exploited (Srinivassan, Ngu, Gedeon, 1994).

3 GENERIC CONCEPTUAL SCHEMA FOR CIS

3.1 Philosophy

The problem of legacy systems is that much of the semantics of a system resides in the application code rather than in the conceptual schema of the system. This therefore hinders communication between pre-existing systems. Irrespective of the approach, whether integrated (Litwin, Abdellatif, 1986), federated (Sheth, Larson, 1990) or multidatabase (Kent, Ahmed, et. al, 1992) taken for interoperability, the designers are faced with the problem of comparing the information content of various databases that need to share information. One possible way out of this is to assume that there exists a DBA with domain knowledge who knows about these commonalties and can identify them (Spaccapietra, Parent, 1991). We confirmed this assumption by conducting an empirical study which consisted of manually comparing the schema of 5 existing CIS in a leading hospital. The systems studied included relational, hierarchical and flat file systems. The study highlighted a number of issues:

- A large amount of knowledge such as rules and constraints on data domain, and application specific knowledge is inherent in the local systems. Some of this knowledge apparently gets represented in the form of entity/object definitions, attributes/properties of objects and relationships among entities/objects.
- The CISs observed are dynamic and undergo frequent structural evolution due to changes in business rules, technology and usage.
- In spite of the resulting structural differences, a medical informatics domain expert could easily identify a lot of semantic similarity among the existing databases. The database structures as found in the data dictionary of each individual CIS coupled

with common querying patterns of users seems to help the expert in identifying semantic similarity.
- This semantic similarity subsequently helps the domain expert in identifying a number of common concepts that exist across these heterogeneous systems.

Given this scenario, we propose an approach of developing an integrated conceptual schema by discovering common concepts in much the same way as the domain expert, by creating meaningful classifications of concepts using data pertaining to structure and usage. The data used for automated classification is gathered from two sources. The first from data dictionary of each CIS and the second by setting up an audit trail that captures common querying patterns of users.

3.2 Approach based on Structure, Usage, Domain Knowledge

Stead and Hammond (Stead, Hammond, 1985) have classified data stored in medical records into the following groups: 1) Demographic data identifying socio-economic data, 2) General non-time-dependent data such as tumour registry information, special patient needs, 3) Problem or diagnosis and associated treatment 4) Time-oriented information such as physical findings, patient's complaints, disease progress 5) Data resulting from orders such as lab tests, and x-rays, 6) Therapies including diets, physio, etc 7) Encounter information such as admission data, including dates, places, care provider, etc (RSP94). Our own empirical study of existing CIS in a large teaching hospital reveals such similar groups of data organized into appropriate entity/objects/file definitions.

The cause of semantic heterogeneity has been discussed extensively in literature (Litwin, Mark, Roussopoulos,1990) (Sheth, Gala, 1989) (Siegel, Madnick, 1991b). Semantic similarity manifests itself in the database structure in the form of entity/object definitions, relationships among entities/objects. The database entity definitions and relationships indirectly reflect the background knowledge of the user groups that specify and use the system. Users belonging to a specific user group from the same application domain are equipped with similar background knowledge, examples of such user groups being doctors, nurses, allied health professionals, and health administrators. The commonality of purpose of these different groups of users results in updates of similar entities as well as similar querying patterns. In addition, usage frequency of some queries also indicate preference of some entities over others.

We assume that similar domain knowledge combined with common objectives leads to the design of similar database structures in heterogeneous databases and that usage pattern carries a cognitive load that indicates background knowledge of users. Database structures can serve as good indicators to identify common generic concepts across CISs and similar usage pattern by a particular user group can indicate similar concepts being used by that user group.

If similar concepts can be derived first from each individual information system and then collectively from the set of information systems that need to interoperate, we have a set of common generic concepts that can be perceived as the integrated conceptual schema of cooperative information systems.

4. CONCEPTUAL CLUSTERING

The purpose of cluster analysis is to place objects into groups or clusters suggested by the data not defined a priori, such that objects in a given cluster tend to be similar to each other in some sense and objects in different clusters tend to be dissimilar. The basic data for cluster analysis is a set of 'N' entities (or objects) on which 'p' measurements (or variables) are recorded. The initial choice of variables reflects the background knowledge of the investigator and the relevance for the purpose of classification (Everitt80).

4.1 Background Knowledge

For our purposes the objects to be clustered are database 'entities' or 'objects' as defined in the actual CISs. The variables chosen are relevant for our purpose as they indicate structure and usage of each entity/object. Each entity/object is characterized by the following variables: (i) *Number of records* (ii*) Number of Attributes/fields/properties* (iii) *Number of keys/links to other entities/objects, (iv) Update frequency (v) Number of users* (vi) *Number of categories of users.* These variables were chosen using some simple heuristic rules used by the informatics domain expert while manually classifying the entities/objects into generic concepts. The other advantage is that they are specific and measurable. Values of these variables play a significant part in clustering the entities into meaningful generic concepts.

The variable *Number of records* serve as a measure to indicate the popularity of the entity. An entity that has a large number of records is used more often possibly by a large number of users. An entity characterized by a large number of attributes is likely to represent an important concept; hence the variable *Number of attributes*. *Number of links* within an entity definition shows the relationship among entities and possibly relationship among concept clusters. A measure of *Update frequency* shows that the entity is being used most frequently, possibly by a large number of users. An entity characterized by a high value of this variable coupled with a high value for the *Number of categories of users* indicates that this entity represents an important concept common to many different categories of users. An entity that is being used by the maximum number of users and has a large value for *Number of records* indicates that the entity is popular among large number of users and is therefore an important concept that is used frequently.

A sample set of data obtained from an audit trail of existing CIS is shown in Table I. The first column of the Table shows the actual names of entities as defined in a particular CIS. The remaining columns show the values of the variables shown as column headings. The method of data collection is explained later in the experimental setup section. The details of the clustering algorithm are explained in the following section[1]

Table1 Structure and Usage data of Clinical Information System

Entity name	***number of records***	***number of attributes***	***update frequency***	***number of links***	***number of users***	***number of user categories***
HCCASE	28769	25	120	3	15	3
HCCONDITIONS	281	4	30	0	5	2
HCLINK	531	3	30	0	5	1
HCPMI	28396	36	80	9	10	2
HCPOSTCODE	2173	2	20	0	5	3
HCPROCEDURES	39	2	10	0	5	3
HCPRINTERS	4	3	5	0	3	3
HCSTAFF	367	11	20	1	5	2
HCTEAM	1661	6	30	1	10	3
HCUSERDEFS	23	8	5	1	1	1
HCUSER	22	6	5	1	1	1

4.2 Clustering Algorithms

A large number of clustering algorithms have been developed with different definitions of clusters and similarity among objects. Hartigan (Hartigan, 1975) and Everitt (Everitt, 1980) are authoritative references on clustering techniques. Clustering techniques are broadly classified into the following 5 types:

1. Hierarchical techniques in which clusters themselves are organized into further clusters, the process being repeated at different levels to form a tree.
2. Optimization techniques in which the clusters are formed by the optimization of a clustering criterion. The clusters are mutually exclusive, thus forming a partition of the set of entities.

[1] Appendix-A includes data from two other CISs that were used for clustering.

3. Density or mode seeking techniques in which the clusters are formed by searching for regions containing a relatively dense concentration of entities.
4. Clumping techniques in which the clusters or clumps can overlap.
5. Others - methods that do not fall clearly into any of the four previous groups.

For our experiment we have chosen an optimization technique as we have a fairly good idea of the possible number of clusters into which the entities of a CIS database can be grouped. Optimization techniques are so called because they seek a partition of the data which optimizes some pre-defined numerical measure, values of which are indicative of a desirable clustering solution.

Using a partitioning algorithm, the observations are divided into clusters such that every observation belongs to only one cluster. The SAS® (SAS, 1989) procedure FASTCLUS that is used by our method is based on Hartigan's Leader Algorithm (Hartigan, 1975) and MacQueen's k-means algorithm (Everitt, 1980). FASTCLUS uses an iterative algorithm for minimizing the sum of squares distance from the cluster means. A set of points called cluster seeds are first selected as a first guess of the means of the clusters. Each observation is assigned to the nearest seed to form temporary clusters. The seeds are then replaced by the means of temporary clusters and the process is repeated until no further changes occur in the clusters. Optionally, based on domain knowledge it is possible to pre-specify the possible number of clusters, in which case the number of iterations may be reduced.

4.3 Experimental Framework

The main experiment is performed in two phases. The first phase consists of running the clustering algorithm using data from each individual CIS in order to obtain concept clusters *within* each CIS. The second phase of the experiment is to run the clustering algorithm using data from all the CIS collectively to obtain clusters *across* CIS. The quality of clusters generated by the algorithm in each case is compared to manual concept classification as a simple means of evaluating the quality of clusters generated by the algorithm. Table I shows a sample data set. The column headings are the variables used by the clustering algorithm. (Appendix-A has some more data used in the experiment). The method of collecting the required data is explained in section 4.3.1.

Data Collection

The CISs used in the experiment are implemented on different platforms such as VAX® cluster and Personal Computer Networks, running under different Operating Systems such as VMS® and Windows® , using different network protocols such as Decnet® , Pathworks® Netbui® and TCP/IP® . Application software environment includes 3 GL languages such as MUMPS® , and COBOL® , 4GL languages such as POWERHOUSE®, Database Management Systems such as CRS® (Clinical Reporting System) and Dbase®. This variety of heterogeneity offered quite a challenge while

setting up the process for data collection. Actual data used in the clustering process was gathered at three different levels: a) data relating to structure such as *Number of attributes* and *Number of links* was gathered from the data dictionaries of each individual CIS, b) transaction logs were set up at the operating Systems level to gather data relating to usage such as *Number of users* and *Number of categories of users*, c) transaction logs at the database level were set up to gather data such as *Update frequency* and *Number of records*. Finally all of this data was moved to the NT advanced Server on which SAS® procedures and programs were tested.

Pre-processing Data

It is often useful to process data prior to the actual clustering. This becomes essential when the variables have (a) different units of measurement, (b) take on different ranges of values and (c) vary with time. For example, the variable *Number of records* will be vastly different for a system that has been in operation for 3 years as compared to a systems that has been in operation for less than a year. Hence this particular variable value cannot be used as it is. However since it is an important variable to be used in clustering, we have ranked this variable on a 10 point scale such that the largest value of this variable has a rank of 10 in each CIS and the lowest value has a rank of 1. All other values are ranked within this range. A larger ranking scale may provide greater accuracy. We use a SAS® procedure called ACECLUS that is useful for pre-processing data to be subsequently clustered by the FASTCLUS procedure. This procedure outputs a data set containing canonical variable scores to be used in the cluster analysis proper.

The Experiment

Initially a variety of options and clustering procedures were tried while choosing the initial cluster seeds After experience with initial runs, selecting good cluster seeds manually was relatively straightforward, and gave us clusters which matched very closely with the generic concepts identified manually by the expert. In future work we would be looking at ways of selecting the cluster seeds automatically so as to reduce the reliance on expert human intervention. Table II shows some comparative figures on the number of clusters obtained using the algorithm, and the number of generic concepts identified manually by a medical informatics domain expert for each individual CIS. The first column indicates the name of the CIS used in the experiment. (This is synonymous with the name of the corresponding clinical department using the system). The number of observations indicate the actual number of entities/objects and their relationships defined in each CIS. The next column shows the number of clusters formed by using the algorithm separately on each CIS. The last column shows the number of concepts identified separately for each CIS by the medical informatics domain expert.

Table II: Comparison of Automated vs Manual Clustering

CIS	*Number of Entities/ Objects*	*Number of clusters generated by the Algorithm*	*Number of generic concepts identified manually*
Radiotherapy	42	9	13
Endocrinology	38	10	12
Community Health	15	8	8
Allied Health	39	10	11
Intensive Care	12	8	9

As can be seen, the number of concepts identified by the human expert seems to be close or marginally higher than the number of clusters generated by the clustering algorithm. One possible reason could be that the manual clustering process could also identify the hierarchical clusters in addition to the disjoint clusters. For example, this was particularly true for systems that had a large number of code files. Since the algorithm at this stage only performs disjoint partitions it lacks the level of sophistication that is possible in manual clustering. Another interesting fact that came to light was that the discrepancy between manual and automated clustering seemed much lower for well designed systems where data is properly normalized, possibly because the well defined entites could themselves represent concepts.

Upon careful examination of the clusters generated by the system, the medical informatics domain expert could label the clusters with names of some familiar concepts. For example in Table A of Fig 1, the entities DIET-TREATMENT, PATIENT-DIET, DRUG-TREATMENT and PATIENT-PROGRAM grouped into cluster 5 by the algorithm could all be classified as belonging to a generic concept labeled **Treatment Plan** by the human expert[2] . Labeling the clusters generated by the system with some generic concept proved to be a fairly easy exercise for the human expert. The 3 Tables of Figure 1 show the results of the clustering algorithm performed individually on three different CISs along with the generic concepts labeled manually by the domain expert.

During the manual concept classification process, although the human expert could identify the broad generic concepts represented within each CIS quite easily, it was too tedious to slot each and every entity into an appropriate generic concept. This becomes quite unmanageable in poorly designed systems. In this regard, the automated system grouped all entities into one cluster or another, and it was an easy feat for the human expert then to label the clusters appropriately and in some cases identify the entity that did not seem to belong to that cluster. For example, in Table A of Figure 1, Cluster 1 represents **Patient Demographics**, Cluster 2 represents **Patient**

[2] The generic concepts identified by the human expert are shown in bold and the actual entity names that are clustered by the algorithm are shown in capitals.

visits, Cluster 3 represents **Test results**, Cluster 4 represents **Treatment Plan**, and Cluster 5 represents the set of **Code files**.

Table: A: CIS I: Endocrinology System

Cluster	Entities	Generic Concept
1	PATIENT	**Patient Demographics**
2	VISIT	**Patient Visits**
3	PATH-RESULTS	**Test Results**
4	CLINIC_ASSESMENT	
5	DIET-TREATMENT PATIENT-DIET DRUG-TREATMENT PATIENT-PROGRAM	**Treatment Plan**
6	LANGUAGE STAFF-CODE RELIGION DOCTOR ETHNIC-GROUP PROGRAM-TOPICS	**Code Files**

Table B: CIS II: Allied Health Information System

Cluster	Entities	Generic Concept
1	PATIENT	**Patient Demographics**
2	PATIENT_ACTIVITY	**Patient Visits**
3	REFERRAL REFERRED_OUT_FOR REFERRAL_SOURCE	**Referral details**
4	THERAPY-COURSE	**Treatement Plan**
5	DEPARTMENT-ACTIVITY	**Staff activity**
6	DEPT_CODE LANGUAGE_CODE THERAPIST_CODE COMPENS_STATUS CLINIC_AREA_CODE	**Code Files**

Table C: CIS III: Community Health System

Cluster	Entities	Generic Concept
1	HCPMI	**Patient Demographics**
2	HCCASE	**Patient Visits**
3	HCCONDITIONS	**Diagnosis**
4	HCPROCEDURES	**Treatement Plan**
5	HCTEAM HCSTAFF HCPOSTCODE HCUSER	**Code Files**

Figure 1 Concept Clusters of Clinical Information Systems.

Having ascertained that the clusters generated by the algorithm matched very closely with generic concepts identified manually, in the second phase of the experiment clustering was performed collectively on data gathered from multiple heterogeneous CISs. In the first few experiments, the resulting global clusters did not group corresponding entities of clusters identified during the individual CIS clustering phase of the experiment. This indicated that similar concepts were not being grouped together to form one generic concept cluster across multiple CISs. Upon further investigation and discussion with the users, we found that in a real life situation, there was no need for all CIS to interoperate. Only some clinical departments manually exchanged data regularly. For example, the diabetes patients treated by the Endocrinology department are regularly referred to the Allied health departments such as Dietetics and Occupational therapy which leads to a regular flow of information between these departments. Similarly, there was a need for the Radiotherapy department to interoperate with the department of Nuclear medicine and Radiation Oncology, but not with Endocrinology. Hence it is possible that our initial

experiments in combining systems together randomly, did not yield any worthwhile concept clusters, since CISs were clustered together arbitrarily, rather than meaningfully.

Our next attempt was to group only those CISs where the respective CIS users indicated a need for Interoperability. The resulting clusters were much closer to the original generic concepts of the participating CISs.

Table A in Figure 2 shows the clusters formed by combining data from Endocrinology system and Community Health Systems which do not manually interoperate. The examples shown are the actual clusters generated by the algorithm. Only a representative sample has been chosen for discussion purposes. To evaluate the quality of the clusters formed, we have used a simple Decision Table as shown in Table B. Currently we do not have a formal procedure in place to evaluate the quality of clusters formed. Our evaluation is based on a human expert's evaluation of the clusters. As shown in the Decision Table (Table B) of Figure 2, out of 6 clusters only 2 clusters (i.e. clusters 1 and 6) generated by the algorithm match with the human expert's generic concept. Clusters 4 and 5 represent the same generic concept as far as the human expert is concerned. Clusters 2 and 3 do not represent any single generic concept as per the expert. So in this case we can say that roughly there is a 33% success as far as the algorithm is concerned.

Table A: Clusters formed with Endocrinology and Community Health Systems

Cluster	Entity	**Generic Concept**
1	PATIENT (I)	**Patient**
	HCPMI (III)	**Demographics**
2	CLINIC_ASSESMENT(I)	**?**
	HCSTAFF(III)	
3	HCCONDITIONS(III)	**Treatment (?)**
	DIET-TREATMENT(I)	
	PATIENT-DIET(I)	
	DRUG-TREATMMENT(I)	
4	VISIT(I)	**Patient Visits**
5	HCCASE(III)	**Patient Visits**
6	LANGUAGE(I)	**Code Files**
	STAFF-CODE(I)	
	HCPOSTCODE(III)	

Table B: Decision Table

Human→ Algorithm ↓	Yes	NO
Yes	2	2
No	2	-

Fig 2 Concept Clusters produced with manually non-interoperating systems.

Fig 3A shows the clusters formed by combining data from Endocrinology and Allied Health Systems, where there is a need for interoperability. In this example 6 out of 8 clusters coincide with the generic concepts identified manually showing an 80% success. The algorithm however has identified 2 separate clusters for **Patient Demographics.** As per the human expert cluster 1 and 2 represents the same generic concept. This is a discrepancy which seems difficult to account for.

These two experiments confirm our hypothesis that common objectives give rise to similar concepts. Although it may be difficult to produce perfect clusters that match with all the generic concepts, the process still yields a large number of generic concepts across systems designed with similar objectives.

Table A: Clusters formed with endocrinology and allied Health Systems

Cluster	Entity	**Generic Concept**
1	PATIENT (I)	**Patient Demographics**
2	PATIENT (II)	**Patient Demographics**
3	VISIT (I)	**Patient Visits**
	PATIENT_ACTIVITY (II)	
4	DIET-TREATMENT(I)	**Treatment Plan**
	PATIENT-DIET (I)	
	DRUG-TREATMENT (I)	
	PATIENT-PROGRAM (I)	
	THERAPY_COURSE (II)	
5		**Test Results**
	PATH-RESULTS (I)	
6	DEPARTMENT-ACTIVITY (II)	**Department activity**
7	CLINIC_ASSESMENT (I)	**Clinical Assessment**
8	LANGUAGE	
	STAFF-CODE	
	RELIGION	
	DOCTOR	
	ETHNIC-GROUP	
	DEPT_CODE (II)	
	LANGUAGE_CODE (II)	
	THERAPIST_CODE (II)	
	CLINIC_AREA_CODE (II)	
	COMPENS_STATUS (II)	

Table B: Evaluation Table

Human→ Algorithm ↓	Yes	NO
Yes	6	1
No	1	-

Figure 3 Concept Clusters produced with manually interoperating systems.

The major advantage of this concept clustering approach is that once identified, these concept clusters, labeled as appropriate generic concepts, can serve as the starting point in providing a customised external view to different categories of users. This is possible because basically the concept clusters are generated using usage patterns. If an audit trail of each CIS usage could be established and fed back to the clustering algorithm one could speculate that it may be possible to greatly improve the quality of these concept clusters. With continuos usage over a period of time, one could hope to arrive at a generic conceptual schema that could reflect the concepts embedded in the underlying systems. This in turn could be the basis for providing highly customised external view across heterogeneous cooperating information systems. Details of providing such a customised external view are discussed in (Srinivasn, Nga, Gedeon, 1994). A hypertext interface that could serve this purpose is presented in (Lee, Ngu, Srinivassan, 1994).

5. CONCLUSION

The approach to integration presented in this paper is based on identifying some common generic concepts that exist across different clinical information systems. A

statistical clustering algorithm is proposed using an audit trail approach to identify these common generic concepts. The algorithm uses existing database structures and usage patterns to find meaningful concept clusters. Experiments performed using this approach have produced encouraging results for clinical information systems that need to interoperate.

The generic concepts represented by these clusters can be used to provide a flexible customised view to each user. Over a period of time with continuous usage monitoring and feed back a customised external view for each category of user can be generated.

Currently, there is no formal mechanism to evaluate the quality of clusters and are instead relying on human intervention to evaluate them. We are also looking at the possibility of providing continuous monitoring and feedback and develop this into a self learning system for automatic concept generation.

Acknowledgments

We acknowledge John Shepherd, R. Bhaskar and Jim Franklin for helpful discussions. We also acknowledge the help provided by various doctors at Eastern Sydney Area Health Service, and to members of the Information Systems Unit at Eastern Sydney Area Health Service for providing the facility to gather the large amount of data required for the experiment.

References

Ahmed, R. Et al. (1991) The PEGASUS heterogeneous multidatabase system. IEEE Computer, 24(12):19-27.

Brodie, M. (1992) The promise od distributed computing and the challenges of legacy information systems. In IFIP DS-5 Semantics of Interoperable Databases Systems: Lorne, Australia,

Bhaskar, R. (1995) Personal Comment.

Collet, C. Huhns M.N.and Shen,W.(1991) Resource Integration using a large Knowledge Base in Carnot. IEEE Computer, vol 24(12): 55-62.

Everitt, B. (1980) Cluster Analysis, Second Edition. Halsted Press, Division of John Wiley & Sons.

Frawley, W.J. Piatetsky-Shapiro, G. Matheus, C.J. (1991) Knowledge Discovery in Databases: An Overview in Knowledge Discovery in Databases, AAAI Press.

Gonzalez, A. Myler, H.,et al. (1991) Automated Knowledge Generation From a CAD Database, Knowledge Discovery in Databases, AAAI Press.

Hartigan, J.A. (1975) Clustering Algorithms. John Wiley and Sons. New York.

Johannesson,P. and Hasan Jamil,M. (1994) Semantic Interoperability Context, Issues and Research Directions. In CoopsIS-94 Proceedings., pp 180-190

Kent,W. Ahmed R. Et.al. (1992) Object Identification in multidatabase systems. In IFIP DS-5 Semantics of Interoperable Databases Systems: 302-319, Lorne, Australia.

Kim. W, and Seo, J. (1991) Classifying schematic and data heterogeneity in Multidatabases systems. IEEE Computer, Vol 24(12): 12-18,

Litwin,W. and Abdellatif, A.(1089) Multidatabase interoperability, IEEE Computer, Vol 19(12):10-18.

Litwin, W., Mark,L., and N.Roussopoulos.(1990) Interoperability of multiple autonomous databases. ACM Computing Surveys, 22(3): 267-293.

Lee, N. Ngu, A.A., and Srinivasan, U. (1994) A Hypertext approach to Querying Clinical Multidatabases. In Proceedings of the 5th International Hong Kong Computer Society Workshop on Next Generation Databases Systems, Honk Kong.

Milliner, S. and Papazoglou, M. (1994) Reassessing the Roles of Negotiation and Contracting for Interoperable Databases. Intelligent Workshop on Advances in Databases and Information Systems, Moscow.

Michalski, R.S and Stepp, R.E. (1985) Learining from Observation: Conceptual Clustering. In Machine Learning An Artificial Intelligence Approach Volume I. Morgan Kaufmann Publishers, Inc. California.

Meersman R. (1994) Personal comment during talk given at University of New South Wales 1994.

Papazoglou, M.P., Laufmann, S. C., and Sellis, T.K. (1992) An organizational framework for Cooperating Intelligent Information systems. International Journal of Intelligent and Cooperative Information Systems, Vol.1,No.1 169-202.

Quinlan J.R. (1990) Learning Logical definitions from Relations. Machine learning, 1(1): 81-106.

Ramirez, J.C.G., Smith, L.A., and Peterson, L.L. (1994) Medical Information Systems: Characterization and Challenges. SIGMOD Record, Vol. 23, No.3.

SAS Institute Inc. (1989) *SAS/STAT* ® User's Guide, Version 6, Fourth Edition, Volume 1, Cary, NC:SAS Institute Inc.

Saltor, F., Castellanos, M.G., and Garcia-Solaco, M. (1992) Overcoming Schematic Discrepancies in Interoperable Databases. In IFIP DS-5 Semantics of Interoperable Database Systems, 272-301.

Piatetsky-Shapiro, G. and Frawley, W. J.(1991) Knowledge Discovery in Databases. AAAI Press/MIT Press.

Sheth, A. and Gala,S. (1989) Attribute Relationships: An impediment in automating Schema integration. NSF Workshop on heterogeneous databases Systems, Chicago.

Stead, W.W. and Hammond, W.E. (1985) Computer Based Medical Records: The Centrepiece of TMR. M D Computing 5(5) p48.

Sheth, A. and Kashyap, V. (1992) So far (Schematically), yet so Near (semantically). In IFIP DS-5 Semantics of Interoperable Database Systems, 272-301, Lorne, Australia.

Sheth, A. and Larson, J.(1990) Federated database systems for managing distributed, heterogeneous and autonomous databases. In ACM Computing Surveys, Vol 22(3).

Stepp, R.E. and Michalski, R.S.(1986) Conceptual Clustering: Inventing Goal Oriented Classifications of Structured Objects, in Machine Learning An Artificial Intelligence Approach, Volume II, Morgan Kaufmann Publishers, Inc.

Siegel, M. and Madnick, S.E. (1991) A Meatadata approach to resolving semantic conflicts. In Proceedings of the 17th International Conference on Very Large Databases, 133-145.

Siegel, M. and Madnick, S. (1991) Context Interchange: Sharing the Meaning of Data, SIGMOD record vol20(4), 77-78.

Srinivasan, U. and Ngu, A.A. (1995) Information Re-engineering in Co-operative Clinical Information Systems. Working Papaer, School of Computer Science and Engineering, University of New South Wales, Australia.

Srinivasan, U., Ngu A.H.H. and Gedeon, T. (1994) Discovey of Generic Concepts from Heterogeneous Clinical Information Systems. Accepted paper for presentation at Second Singapore International Conference on Intelligent Systems (SPICIS'94).

Spaccapietra, S. and Parent, C. (1991) Conflicts and correspondence Assertions in Interoperable Databases. SIGMOD record special, vol 20(14): 49-54.

Siegel, M. E. Sciore, S. Salveter. (1991) Rule Discovery for Query Optimization, Knowledge Discovery in Databases, AAAI Press.

Sciore, E. Siegel, M., Rosenthal, A. (1991) Using Semantic Values to Facilitate Interoperability Among Heterogeneous Information Systems. ACM Transactions on Database Systems, Volume 19 No 2.

Zytkow J M and Baker, J. (1990) Interactive mining of Regularities in Databases. Knowledge discovery in Databases, AAAI Press.

Appendix-A

Table II: Structure and Usage data of Clinical Information System-II

Entity name	number of records	number of attributes	update frequency	number of links	number of users	number of user categories
CLINICAL-AREA-CODE	146	5	5	1	10	1
CONTACT-TYPE-CODE	158	5	5	2	10	1
COMPENSIBLE-STATUS-C	10	3	5	0	10	1
CLINICAL-STAT-CODE	17	5	5	1	10	1
CODE-TYPE	24	3	5	0	10	1
DEPR-ACTIVITY	6402	7	50	3	50	2
DEPT-ACT-CODE	308	5	5	2	10	1
DEPT-CODE	5	3	5	0	10	1
DOCTOR-CODE	374	3	5	0	10	1
FIN-CLASS-CODE	74	3	5	0	10	1
GRP-ACTVTY-CODE	68	5	5	2	10	1
GENERIC-CODE	677	4	5	0	10	1
GRP-CLN-THERAPIST	503	8	40	3	10	1
HOSP-GROUP	2	4	5	0	10	1
HOSPITAL	10	8	5	1	10	1
INSURANCE-COVER-CODE	7	3	5	0	10	1
INTERVENE-CODE	1139	5	5	2	10	1
LANGUAGE-CODE	92	3	5	0	10	1
LOCAL-CODE	7614	6	5	2	10	1
MARITAL-STATUS-CODE	7	3	5	0	10	1
MEDI-DIAG-CODE	2638	5	5	2	10	1
PAT-ACTIVITY	27816	16	80	4	50	2
PAT-THER-GROUP	1018	9	20	4	20	2
PAT-STATUS-CODE	3	3	5	0	10	1
PATIENT	3968	23	120	0	50	2
REFD-OUT-FOR-CODE	75	5	40	2	10	1
REFERRAL	5582	32	120	3	50	2
REFERED-FOR-CODE	133	5	40	2	10	1
REFERRAL-OUT	177	9	40	4	40	2
REFERAL-SOURCE-CODE	295	5	40	2	20	2
REFD-OUT-TO-CODE	281	5	40	2	10	2
THERAPY-COURSE	5044	11	120	4	50	2
THERAPY-DIAG-CODE	2451	5	80	2	50	2
THERAPY-GRP-CLINIC	402	13	40	4	20	2
THERAPIST	ATH	1	191	7	40	10
THERAPY-GRP-CODE	160	7	40	3	0	0

THER-DISCH-REASON	72	5	40	2	10	1
THERA-DISCH-TO-CODE	101	5	40	2	10	1
WARD-ID	109	2	5	0	10	1

Appendix-A

Table I: Structure and Usage data of Clinical Information System-I

Entity name	number of records	number of attributes	update frequency	number of links	number of users	number of user categories
CLINICAL-ASSESSMENT	500	5	20	3	10	5
CHO-EXCHANGE	100	10	25	3	5	3
	10	2	10	1	2	2
COUNTRY	100	2	10	1	2	2
DIET-ADV-GROUP	10	2	10	1	2	2
DIABETES-CONTROL	50	4	20	2	5	2
DRUG-DOSAGE	400	8	40	5	10	4
DRUG-TREATMENT	200	4	30	4	10	4
DOCTOR-DETAILS	50	11	30	3	5	3
DRUGS	50	5	20	2	5	2
DOSAGE	20	3	20	1	5	2
DIET-TREATMENT	200	9	20	3	10	3
ETHNIC-GROUP	50	2	10	1	2	2
HOSPITAL-ADMISSIONS	100	3	20	2	5	4
HOSPITAL-LOCATION	10	2	10	1	2	2
LANGUAGE	50	2	10	1	2	2
PATIENT	500	80	40	18	20	10
PROGRAM-BOOKINGS	400	5	30	3	5	2
POST-CODE	100	2	10	1	2	2
PATIENT-DIET-RECORD	200	4	30	4	10	4
PAT-EXCERCISE	100	5	30	3	5	3
PROGRAM-FOLLOWUP	50	3	20	1	5	3
PAT-INSURANCE	300	2	20	2	5	3
PATIENT-PROGRAM	200	3	30	2	10	4
PATH-RESULTS	1500	5	40	3	15	4
PROGRAM-SCHEDULE	50	4	20	3	5	2
PROGRAM-TOPICS	50	2	20	2	5	2
PAT-TREATMENT	400	5	30	4	5	2
REVIEW-BOOKINGS	200	5	30	2	5	3
RELIGION	50	2	10	1	2	2
REASON-FOR-VISIT	20	2	10	1	4	2
SPECIALITY-CODE	10	2	10	1	2	2
SOURCE-OF-REFERRAL	20	2	10	1	4	2
STAFF-CODE	50	2	10	1	2	2
TEST-CODE	100	5	20	3	10	5
TREATMENT-GROUP	100	2	20	1	5	3
TYPE-OF-TREATMENT	50	2	10	1	5	3

VISIT	500	40	40	8	15	4