Mälardalen University Licentiate Thesis
No.35

Clustering and Case-Based Reasoning for User Stereotypes

Mikael Sollenborn

October 2004

MÄLARDALEN UNIVERSITY

Department of Computer Science and Engineering
Mälardalen University
Västerås, Sweden

ISSN 1651-9256
ISBN 91-88834-66-2
Printed by Arkitektkopia, Västerås, Sweden
Distribution: Mälardalen University Press

Abstract

A user stereotype represents a certain kind of user who exhibits a set of specific characteristics - an abstraction of a group of similar users. Looking at statistical data gathered from a population of users, these groups can be identified either manually or using automated clustering techniques, and constructed by generalizing the most significant features of the identified groups. Case-Based Reasoning (CBR) is an Artificial Intelligence (AI) method based on the idea of reusing past experience in a domain-specific library of problem-solution descriptions, known as cases. By representing a solution to the problem of supplying a typical kind of user with appropriate information, it is natural to see user stereotype cases as part of a CBR process. This thesis describes the usage and creation of user stereotypes in novel domains, aided by the use of clustering techniques. The first application domain is personalization on the World Wide Web (WWW), where user stereotypes and a filtering technique called category-based filtering are combined to handle a frequently occurring problem on WWW sites attempting to automatically recommend items of interest to site visitors. In the second application domain, psychophysiological medicine, clustering is utilized to identify recurring patterns of previously classified time-series of Respiratory Sinus Arrhythmia (RSA). Using a combination of expert knowledge and repeated clustering, the aim is to incrementally build a case library of stereotypes which can be used in a CBR system for automated classification of RSA sequences.

To my family

Preface

Finally, it is finished! Didn't think that would happen.

Without the support of a number of people, these words would not have seen the light of day. First of all, I owe many thanks to my supervisor Peter Funk for countless hours of guidance and discussions. I would also like to thank my two sponsoring companies that made my research possible, Eyescream AB and PBM Stressmedicine, as well as my assistant supervisor in the latter company, Bo von Scheele.

Sharing the same room at the department as well as supervisor, sponsor, and (more recently) project, for more than two years, Markus Nilsson has been a valuable colleague and friend. Always kind of funny, always a pain - thank you.

My special and heartfelt thanks to Annica and Therese, who contributed greatly to make the last couple of years more bearable.

Last but definitely not least, I would like to thank my parents and my family for believing in me and supporting me no matter what I choose to do in life.

Mikael Sollenborn
Västerås, September 23, 2004

Publications

Publications included in the thesis

Mikael Sollenborn and Peter Funk. Category-Based Filtering and User Stereotype Cases to Reduce the Latency Problem in Recommender Systems. In *6th European Conference on Case-Based Reasoning*, pages 395-405. ECCBR'02, Aberdeen, September 2002.

Markus Nilsson and Mikael Sollenborn. Advancements and Trends in Medical Case-Based Reasoning: An Overview of Systems and System Development. In *17th International FLAIRS Conference, Special Track on CBR*, pages 178-183. FLAIRS'04, Miami, May 2004.

Mikael Sollenborn and Markus Nilsson. Building a Case Base for Stress Diagnosis: An Analysis of Classified Respiratory Sinus Arrhythmia Sequences. In *7th European Conference on Case-Based Reasoning - Workshop Proceedings*, pages 55-63. ECCBR'04, Madrid, August 2004.

Publications not included in the thesis

Mikael Sollenborn and Peter Funk. Category-Based Filtering in Recommender Systems for Improved Performance in Dynamic Domains. In *2nd International Conference on Adaptive Hypermedia and Adaptive Web Based Systems*,pages 436-439. AH'02, Malaga, May 2002.

Markus Nilsson, Peter Funk, and Mikael Sollenborn. Complex Measurement Classification in Medical Applications Using a Case-Based Approach. In *5th International Conference on Case-Based Reasoning - Workshop proceedings*, pages 63-73. ICCBR'03, Trondheim, June 2003.

Contents

I

Thesis

Chapter 1

Introduction

In many situations an advantage can be found in making the assumption that despite the individuality of every person, similar behavioral patterns and characteristics can be extracted by studying a population. As a result of such analysis, people can be classified into groups. There is, however, a difference between a group as such, and (user) stereotypes. A stereotype is a representation of a group of people as a single individual, which exists not as a real person, but only as an extraction of the most common features of a group of people. The term "user" can imply several things: it could mean an active user, such as a person browsing the contents of a web site, or a passive user, such as a "user" of treatments as proposed by a physician. Depending on the application and the amount of data available, the construction of stereotypes could be done manually or by automatic extraction of similar features (initiated by merging similar users using clustering), or any combination thereof.

Case-Based Reasoning (CBR) is an AI method for building intelligent systems using the notion of knowledge reuse and problem similarity. In essence, the method is based on comparing new cases with stored knowledge in a case library, and reusing the solution to the most similar case. Employing a CBR perspective, a user stereotype can be seen as a representation of a case containing the solution to the problem of supplying a typical kind of user with appropriate information.

This licentiate thesis will illustrate the usefulness of the combined approach of user stereotypes, clustering, and Case-Based Reasoning, by looking at distinctly separate application domains: web page personalisation in e-commerce, and case library construction for automated classification in the medical domain.

1.1 Thesis outline

The licentiate thesis is organized as follows. Chapter 2 provides a background to the most important methods and techniques used in the thesis. Chapter 3 contains descriptions of background, related work, motivation, and contribution to the two application domains: web page personalisation and phychophysiological medicine. Chapter 4 presents the included papers and summarizes each one. In the fifth chapter we take a look at future work and conclude part I of the licentiate thesis. The following three chapters, 6-8, contain the full version of the previously published papers included in the thesis.

Chapter 2

Background

2.1 Case-Based Reasoning

Case-Based Reasoning (CBR) [1, 2, 3] is both a model of human reasoning, and a method used to create "intelligent" systems. As a model, CBR is based on a number of key observations. The first observation is the fact that most of the problems a decision maker has to handle are not unique. When encountered with a new problem, novices and experts often reason by analogy, comparing the current situation with earlier problems encountered. The second observation is that when solving new problems, people typically reuse solutions from similar problems, adapting the solution to suit the current circumstances. In summary, the CBR model of human reasoning suggest that people reason by analogy, remembering past experiences.

In practice, CBR is an AI method for building intelligent systems based on reuse of past experiences, represented as problem-solution descriptions known as cases. Building a case library covering the area in question is essential. The case library needs to cover a sufficiently large part of the problem space from the start, as adaptations to new problems are often hard to make if there are no stored cases similar enough to the new problem. A case typically consists of a problem description, a set of identifying features, and a solution to the problem.

The CBR problem solving cycle, as illustrated in Figure 2.1, is often referred to as the 4 RE:s: REtrieve, REuse, REvise, REtain. In the first step, Retrieve, the cases most similar to the current problem are selected using some kind of similarity metric, such as Nearest Neighbor or Inductive Retrieval, as described in section 2.1.1. In the retrieval process, weights are typically used to put emphasis on case features that are considered more significant in the comparison. Weights may be assigned according to human expertise, or by some kind of automatic weighting system. In the Reuse step, the most similar of the cases selected in the first step is determined using additional similarity reasoning. If the current problem and the closest matching case are still dissimilar, the solution to the closest matching case is adapted using domain-specific rules. A proposed solution is then presented to the system user. If the suggested solution was inappropriate, a Revision has to be made, based on the error report, which may be manual or automatically inferred. In the last step, the problem along with the solution is Retained in the case base if the current case differed substantially from the closest matching case.

2.1.1 Measuring similarity

Arguably the most important part of any CBR system is the retrieval and matching process. Inherent in this process is the equally important case representation problem, i.e., how to represent a problem as a number of (most often) numeric features. A case consists of variables arranged in a vector formation. The variables represent features describing the vector in an N-dimensional space, hence the name *feature vector*. Every feature adds a new dimension to the N-dimensional space. As mentioned previously, features are often numeric, but may also consist of text, images, or any other type of data. However, in the end, features must be comparable, and a numeric transformation of non-numerical features is usually required before a case similarity matching can be done.

The most common approach for similarity matching in CBR is Nearest Neighbor matching, also known as k-NN. The k-NN algorithm is used to calculate the euclidian distance between two feature vectors, thereby trying to find the k nearest vectors or cases. As previously mentioned, weights are used in the comparison to assign different levels of importance to different features. Equation (2.1) illustrates a typical similarity matching using k-NN. The k most similar cases are retrieved as a result

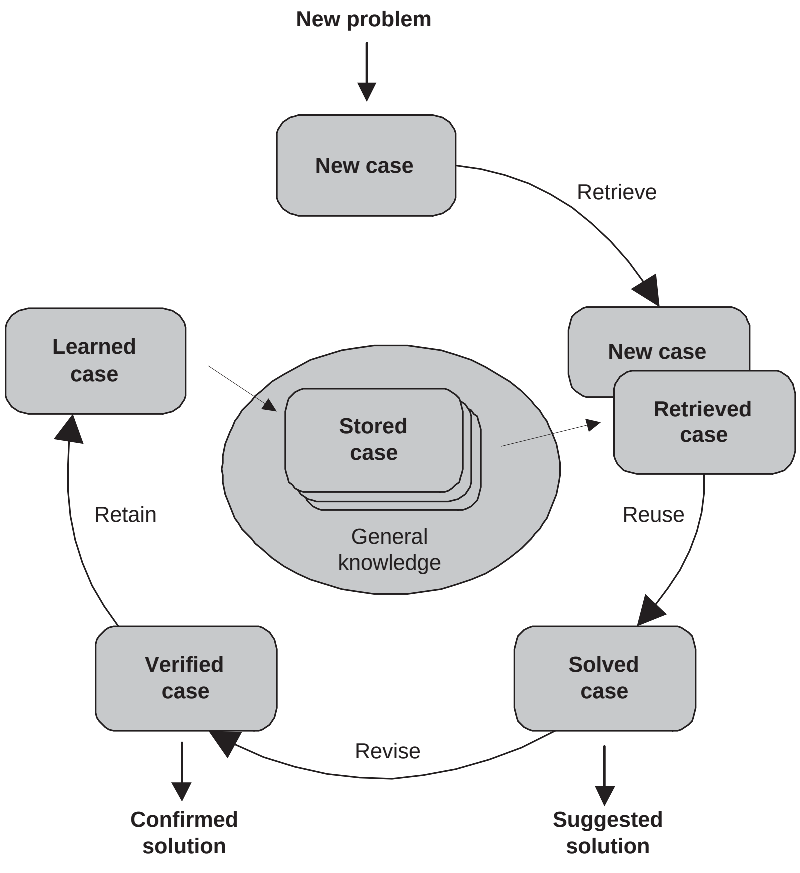

Figure 2.1: The CBR problem solving cycle.

of comparing the feature vector in the new case, nC, with the feature vector in the stored case sC. The most similar cases are then ordered according to their similarity value.

$$similarity(sC, nC) = \sum_{i=1}^{n} W_i \times f_i(sFeature_i, nFeature_i) \qquad (2.1)$$

Other techniques can be used for case similarity matching as well, such as adaptive retrieval [1] and neural networks [4]. A large set of training cases is usually required when a neural network is used as the retrieval technique in CBR. The network also has to be retrained as soon as a new case is added to the case base, which is a disadvantage.

Adaptive retrieval is a method often involved in reducing the retrieval time for CBR systems. As the case base grows, retrieval time increases linearly when using k-NN. Methods like inductive retrieval index the whole case base offline for better retrieval performance. Each case is assigned a position in the N-dimensional problem space, and an index tree is constructed with the indexed cases as basis. The retrieval time decreases since retrieval is reduced to a simple traversal of the index tree. One of the drawbacks of inductive retrieval is that the index tree must be rebuilt every time a new case is added.

2.1.2 Case libraries

The case library, also commonly referred to as the case base, is a database containing all available domain knowledge in the form of cases. When a new case is encountered it is compared to the cases in the case base, and if no similar case could be found the new case is usually retained in the case base together with its (often adapted) solution. Whether or not to retain the new case is often decided by a human expert. For many systems, it is impractical to store all new cases in the case base even if they differ slightly from existing cases, since the case base will quickly grow too large. However, one alternative approach is to retain all cases and perform case library maintenance at a later stage.

There may be a number of reasons that case library maintenance becomes necessary. As mentioned above, the case base may have grown too large to handle efficiently, often resulting in unacceptable retrieval times. There may also be inconsistencies in the case base, resulting in a need to identify and remove invalid cases. Another possible reason is that new features have been added to the case representation, and the old cases must be updated with the new information.

One way of dealing with case libraries that have grown too large is to utilize case aggregation. In this process, cases are clustered into stereotypical cases, or prototypes [5]. A prototype is a generalisation of the cases it represents.

2.1.3 Adaptation

Adaptation is one of the most difficult parts of the CBR cycle and there is no general method that can be used for all or even a majority of CBR systems. In fact, most CBR systems leave out adaptation altogether, or assign the task to a human expert. The need for adaptation arises when a new case is encountered for which there is no satisfactory solution in any of the most similar cases in the case library. For adaptation to be possible, the problem domain and its underlying parameters have to be well understood and documented. Any of a number of adaptation methods may be utilized, such as model-guided repair [1], where the features themselves may be substituted.

2.1.4 Advantages and disadvantages

When discussing the advantages and disadvantages of CBR, CBR is often compared to Rule-Based Reasoning (RBR). RBR was the first AI method used to construct computer-aided decision support systems, later to be known as expert systems. After initial success, RBR eventually ran into a number of problems. One, it required extensive expert knowledge of the domain, not limited to knowing solutions to common problems, but intrinsic in-depth knowledge of the underlying properties and parameters of the domain. Two, maintenance is problematic because to extend or modify the expert systems, rules had to be modified, which in turn could affect other rules, causing the system to malfunction. Thus, one error in the rule base could possibly render the whole system useless. Three, RBR systems are typically capable of delivering only one answer. If the proposed solution is unusable, little can be done to produce alternative solutions, without changing the actual rules.

CBR, to a certain degree, remedies all of these problems. Building a case base that covers the majority of the problem space of a domain usually still involves the help of experts, but the knowledge does not need to be intrinsic. If enough cases can be gathered and appropriately represented, the system can still provide valid answers. Thus, CBR is suited for domains where the domain knowledge is weak. Maintenance is less problematic in CBR since the knowledge is strictly local, i.e. cases are compared to one another but they do not influence each other. Thus, adding an invalid case does not cause overall problems. When providing answers, the fact that the neighbor matching of CBR is not absolute but merely gives an estimation of the similarity of two cases means that if the first proposed solution is inadequate, the solution to the second or third most similar case can often be used instead. It should be noted however, that these advantages does not imply that CBR is necessarily "better" than RBR, it simply means that the methods are suited for different kinds of problems. If the domain is fully understood and can be formalized effectively, an RBR system is often preferable.

CBR also has some general disadvantages. As mentioned earlier on, the case base must cover the majority of the problem domain, or the system will run into problems when cases that have no real match in the case base appears, since no solution can be proposed. If the system must provide an answer (for critical systems etc) the system may be forced to propose a solution to a problem with a very low similarity to the current case. Additionally, CBR systems suffer from inherent unreliability. Although the reliability of a CBR system increases with the proportion of coverage of the problem domain, reliability cannot be guaranteed. Adding new cases will not necessarily make a system converge towards greater reliability, as cases add only local improvement.

2.2 User Stereotypes

Traditionally, the term *user model* represents the current knowledge about a user as an individual, gathered through measurements, questionnaires, observation etc. A *user stereotype*, in contrast, represents a certain kind of user who exhibits a set of specific characteristics. One approach to constructing user stereotypes is manual creation, based on e.g.

age, sex, or any other feature or combination thereof. User stereotypes may also be identified by using clustering techniques to group similar users and identifying the key aspects of their similar features.

One of the primary advantages of utilizing user stereotypes as a complement to (or even as a replacement of) user models is that before knowing a new user to the full extent, a system can make early assumptions about which type of user he/she is, based on the currently available personal information. Thus, qualified guesses can be made regarding which kind of action should be appropriate to satisfy a particular user [6] or strengthen a hypothesis in a medical context.

As introduced by Rich in [7], user stereotypes require two types of information. The system must know what properties capture a stereotype, and what events or behavior that implies a particular stereotype. If this information is highly dynamic and domain dependent, a clustering approach is preferable to static stereotypes, since it is able to automatically identify related categories and adapt to a changing population of users, their preferences and their characteristics.

User stereotypes are in common use among researchers in the User Modeling community. In the early work, user stereotypes were mostly constructed manually [8, 9]. The manual approach to finding user stereotypes is often a difficult task, which involves classification of users by experts, and in-depth analysis of data concerning the interests of the individual users. Such difficulties led many researchers to focus on automatic acquisition of stereotypes using machine learning techniques. A number of automatic learning techniques have been proposed, such as decision trees [10] and neural networks [11].

User stereotypes can also be incorporated into the Case-Based Reasoning methodology. By representing a solution to the problem of supplying a 'typical' kind of user with appropriate information, it is natural to see user stereotypes as cases in a case library. Whenever the information in a single user model is insufficient for deciding which action to take, the user stereotype case most closely resembling the user is consulted to make assumptions about the user's expected behavior (Retrieve, Reuse). The case is revised when the user evaluates the recommended items, and Retained when the user stereotype cases are updated.

In this thesis, the term user stereotype will sometimes be exchanged with terms such as session stereotype and phase stereotype, to indicate more precisely the contents of the stereotype which is being constructed and used. However, the general definition of a stereotype as a generalization and extraction of the most important shared features of a group remains the same.

2.3 Clustering

Clustering is a type of multivariate statistical analysis also known as cluster analysis, or unsupervised classification analysis. The main objective of clustering is to find similarities between samples, and then group similar samples together to assist in understanding relationships that might exist among them. More formally, the object of clustering is to group data points into subsets (clusters) in such a way that the objects within each cluster is more related to one another than objects within another subset. Thus, the notion of similarity is essential to clustering [12, 13].

There are a number of characteristics that distinguish different approaches to cluster analysis.

- *Flat* vs. *hierarchical* representations. In a flat representation, each cluster is a complete group which does not contain subgroups, and the number of clusters is usually determined in advance. In a hierarchical representation, each cluster contains subgroups, each of which in turn can also be divided into further subgroups, recursively.

- *Overlapping* vs. *disjoint* clusters. If the algorithm allows for overlapping clusters, an entity can be part of more than one cluster, whereas the opposite is true for the disjoint approach.

- *Incremental* vs. *non-incremental*. In a non-incremental approach, wishing to add additional entities of data to a previously completed clustering forces a need to re-cluster all entities. An incremental approach to clustering represents an attempt to allow for entities to be added continually, without the need for re-clustering.

- *Agglomerative* vs. *divisive*. Top-down or bottom-up creation of cluster tree. Applicable to hierarchical clustering only and described in section 2.3.2.

To measure similarity, either between cluster members or entire clusters, one of a number of distance metrics is used. Distance metrics are separated into *Distance Measurements Between Data Points*, and *Distance Measurements Between Clusters*. The most common of the first kind are Euclidian distance, Manhattan distance, and Pearson correlation distance. The latter are divided into average linkage, single linkage, and complete linkage.

Three of the most common clustering techniques are k-means, hierarchical clustering, and Self-Organizing Maps (SOM).

A problem inherent in every clustering method is the problem of choosing the optimal number of clusters. Choosing too many clusters compromises generality, but choosing too few clusters may result in less distinct, less informative cluster groups. In many cases, the number of clusters is determined on a trial and error basis, or decided upon using manual expert analysis.

2.3.1 Measuring similarity

Distance Measurements Between Data Points

When measuring the distance between data points [14], one is concerned with comparing pairs of individual members of separate clusters. In practice, the metric gives a numerical value to the amount of dissimilarity between two vectors. Which distance metric to use is often subjective and can be chosen to suit the needs of a specific clustering problem. The most commonly used metric is the Euclidian distance, which is defined in Equation (2.2) below. In the following two equations, n represents the number of vector elements, and x_i and y_i represents the vector element i in vectors x and y, respectively.

$$d(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{2.2}$$

The city block, or Manhattan distance, as shown in equation (2.3), is an alternative often used.

$$d(x,y) = \sum_{i=1}^{n} |x_i - y_i| \tag{2.3}$$

Another commonly used metric is the Pearson correlation coefficient. Often seen variations of Pearson include the Uncentered Pearsons Correlation coefficient and the squared Pearson correlation coefficient. The metrics for the standard Pearson correlation coefficient is shown in equation (2.4) below, where $\overline{x}$ and $\overline{y}$ are the mean value of vectors x and y, respectively.

$$dc(x,y) = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sum_{i=1}^{n}(x_i - \overline{x})^2 \sum_{i=1}^{n}(y_i - \overline{y})^2} \tag{2.4}$$

Distance Measurements Between Clusters

To measure distances between clusters [13], one of three methods is typically used. In the single linkage method, the distance between two clusters is said to equal the minimum distance between any two members of the two clusters, as seen in equation (2.5). The primary disadvantage of this method is the fact that it tends to force clusters together regardless of the positions of other members of the cluster. In the following three equations, r and s are two separate clusters, d is a distance function between data points (as described above), and r_i and s_j are cluster members i and j of clusters r and s.

$$dc(r,s) = Min(d(r_i, s_j)) \tag{2.5}$$

The second method is complete linkage, which measures similarity as the maximum distance between any two members of two clusters. This method, as illustrated in equation (2.6), creates compact clusters but should not be used if there is noise expected in the data.

$$dc(r,s) = Max(d(r_i, s_j)) \tag{2.6}$$

The third, and most commonly used method, is average linkage. Average linkage is the most computationally expensive method of the three, since it calculates the mean distance of all possible pairs of members of two clusters. It avoids the problems of noise and cluster chaining and is thus usually the preferred method. Average linkage is mathematically defined in equation (2.7), where n_r and n_s are the number of members of clusters r and s, respectively.

$$dc(r, s) = \frac{1}{n_r n_s} \sum_{i=1}^{n_r} \sum_{j=1}^{n_s} d(r_i, s_j) \tag{2.7}$$

2.3.2 Clustering techniques

Hierarchical clustering

In hierarchical clustering [15, 13], data is partitioned into a tree-like structure by recursively merging or dividing clusters. An *agglomerative* approach to hierarchical clustering starts with representing each data entity as a unique cluster and then merges these entities into gradually bigger clusters, ending up with all entities belonging to one single cluster. The *divisive* approach starts with a single cluster and divides it into smaller clusters until each cluster is represented by only one entity, or until a distance threshold between clusters is satisfied.

Hierarchical clustering can be illustrated by a two dimensional diagram known as a dendrogram, as shown in Figure 2.2. In this figure, 1-5 represent initial observations (single entities), but could also be seen as clusters with only one member. In the bottom-up (agglomerative) approach, 1 and 2 are initially identified as the two most similar clusters/entities and merged, followed by 4 and 5, etc, until all elements belong to only one cluster.

Agglomerative hierarchical clustering can be divided into the following basic steps:

1. Calculate the distances between all objects.

2. Find the two clusters r and s with the minimum distance to each other.

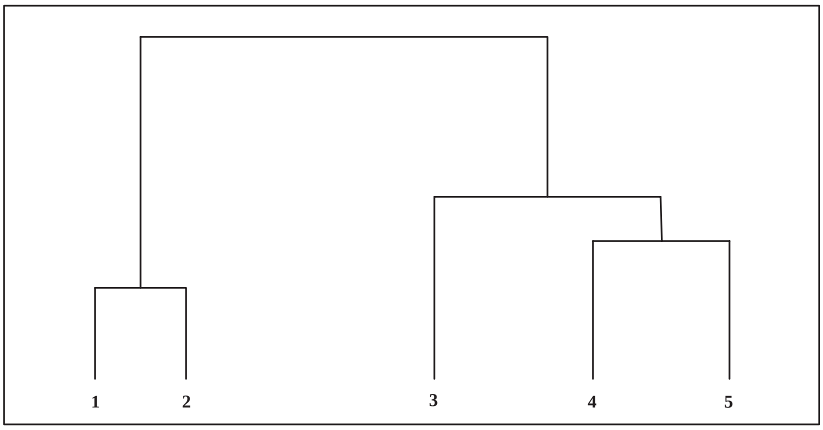

Figure 2.2: A dendrogram showing the hierarchical formation of clusters, starting either from single observations (bottom-up), or from all observations in one cluster (top-down).

3. Merge the clusters r and s and replace r with the new cluster. Delete s and recalculate all distances which have been affected by merging the clusters.

4. Repeat step (2) and (3) until the total number of clusters become one.

K-means clustering

K-means clustering [16, 13] uses a flat representation, and differs from hierarchical clustering in that the number of clusters, k, needs to be predetermined before clustering starts. The goal of k-means clustering is to divide the data into k clusters in such a way that the *loss function* is minimized for each cluster. Since the number of clusters needs to be pre-determined, non-hierarchical clustering techniques such as k-means are better suited if the data is composed of known or at least partly known classes.

The loss function typically used is the sum of distances between a cluster's *centroid* (the mean object in a cluster j) q and the members the cluster j contains $(x_1 \ldots x_n)$, thus

$$L = \sum_{i=1}^{n} d(x_{ij}, q_j) \tag{2.8}$$

In the most commonly used procedure for k-means clustering, each object is initially randomly assigned to one of the k clusters. The positions of the k centroids are determined and objects are reassigned to different clusters based on which centroid is closest to the object. The process is then restarted by determining new centroids followed by another reassignment, until no new reassignments of objects occur.

Self-Organizing Maps

Self-Organizing Maps (SOM), or Kohonen Nets [17] is a neural network approach to clustering, and has become fairly popular in recent years. Its most important advantages are moderate time and memory requirements, and its ability to visually present the result in a meaningful way. Its major disadvantage is that it needs specification of many parameters (typically with unknown values) before the clustering can begin. Among these parameters are the dimensions of the Kohonen net, which in turn decides the (maximum) number of clusters.

Chapter 3

Application domains

In the authors early research, the focus was set on personalization on the WWW, utilized in the domain of e-commerce. After switching research projects and getting a different sponsor, the focus was changed significantly, to psychophysiology in medicine. However, from early on it was clear that many of the ideas used in the first domain could be reused in the second, thus making an attempt to show the versatility of the proposed approach. In this chapter, the application domains are described with respect to background, motivation, related work, and contribution.

3.1 Web personalization

3.1.1 Background

In the computer science domain, personalization is defined as the process of selecting and adapting information to suit individual users' needs. The motivation behind personalization is the idea that programs learning from user preferences will result in more dynamic products, that are easier and more flexible to use. Most work done on personalization has focused on adaptation of network-distributed information, such as electronic mail, news group postings, and especially World Wide Web sites. On the WWW, so called Recommender Systems have been extensively used for personalization of web sites involved in e-commerce. The Recommender Systems functions by presenting individually selected data items to web site visitors based on their previous choices. The expected result is that users will tend to buy more because of the personalization system's automatic suggestions of interesting items, and that the user generally will feel more happy about the visit and be willing to come back later as a result of the personal treatment. Often, the term personalization is also used in conjunction with network based virtual communities, where individual users explicitly rate items to help the underlying recommendation engine suggest items by finding users with similar preferences and goals [18, 19, 20].

Closely related to the term personalization, and especially personalized data, is information filtering. Also commonly referred to as "data mining" and "SDI" (Selective Dissemination of Information), information filtering can be described as the process of sorting through large amounts of data, selecting information of interest according to filtering criterions. The most commonly used criterion is the preferences of a system user.

Information filtering can be broken down into three subtasks, as seen in Figure 3.1.1. The first step is the collection of the information sources. Then, the selection, or filtering, takes place. The last step in this process is presenting the selected information to the end user [21, 22].

Figure 3.1: The information filtering process.

Content-based filtering is the earliest of the "classic" filtering techniques. Essentially, it is based on comparing user preferences with available data items. In content-based filtering, each user is an island, because no comparisons with other users are made. Instead, each user is supposed to display his preferences in one way or the other, enabling the system to filter information using the preferences. In early systems, the preference selection was often made by requiring each user to enter keywords into his/her profile.

Collaborative filtering is based on the idea of 'peer review' recommendations. Instead of comparing user preferences with data items, the user is compared to other users. Users with similar preferences are identified and grouped, and items that many of the similar users has seen and rated positively are chosen and presented to the current user [22, 23].

3.1.2 Motivation

The *latency* problem is a frequently encountered obstacle for Recommender Systems on the WWW. The filtering technique most often used in Recommender Systems, known as collaborative filtering, depends on implicit or explicit peer reviewing of items to enable finding users with similar preferences and recommending items based on those similar users. The latency problem represents the difficulty of introducing new, unrated items into such a system. Since new items have not yet been rated by any user, they are not included in any recommendations, which leads to few users getting a chance to rate them, and thus, the problem persists. The latency problem is especially noticeable if the information on the web site is highly dynamic and new items are added frequently. In an e-commerce environment new items are often the most important ones to advertise to the user, but the latency problem prevents the system from making valid recommendations of such items. As described in section 3.1.3, some approaches to handling the latency problem have been proposed, but these approaches suffer from various problems.

In Paper A, a novel approach to reduce the latency problem is presented, referred to as category-based filtering.

3.1.3 Related work

Introduced early on by Rich [7] and further developed in [24], user stereotypes have often been employed in the user modelling community, e.g. by Paliouras et.al. [10] to model users in a dialog system, by Jameson [25], applying a psychological perspective, by Chin [26] who explore the advantages of user stereotypes compared to user models, and by Dailey [27], who interestingly use stereotypes as a way of handling bias in statistical data.

The usage of stereotypes is also very common in information filtering on the web, as a way of classifying users. Ardissino and Sestero [28] use stereotypes as a way of modelling user plans. Kuflik *et al.* [29] provide a thorough comparison of information filtering systems based on user stereotypes versus filtering systems based on personal, individual information. Henze and Nejdl [30] utilize stereotypes in an online educational system to provide better learning.

CBR is commonly used in information filtering alongside filtering strategies such as collaborative and content-based filtering. It has been argued by Hayes *et al.* [31] that under certain conditions, collaborative filtering and CBR can be seen as synonymous.

The latency problem, as described above, has been addressed by a few researchers. Content-based filtering in addition to collaborative filtering, as proposed by Funakoshi and Ohguro, may be a solution, but runs the risk of only recommending items almost identical to the ones the user has appreciated before [32]. In [31], Hayes *et al.* points out that a solution to the latency problem is to categorize the items, but does not develop the idea further. Hybrid filtering systems combining content-based and collaborative filtering have also been used for other purposes than reducing the latency problem, e.g. by Cotter and Smyth for efficient TV program recommendations [21].

3.1.4 Contribution

A novel approach to combining collaborative and content-based filtering for reducing the latency problem. Although several systems have explored this combination previously, none have focused on the inclusion of meta-data in the collaborative filtering process; that is, comparing previously categorized items as opposed to comparing representationless items. As a result, the same item data (category belonging) can be used for both content-based and collaborative filtering, and the latency problem can be reduced.

3.2 Psychophysiology in medicine

3.2.1 Background

In the early work of John Stern (1964), psychophysiology was defined as "any research in which the dependant variable (the subject's response) is a psychological measure and the independent variable (the factor manipulated by the experimenter) a behavioural one" [33]. Stern was part of the group of physiological psychologists led by R.C. Davis, who in the mid-1950's established psychophysiology as a separate discipline. Later on, it became clear that the dependent and the independent variable could be switched, studying the effects of manipulating psychological variables to cause behavioral changes. Using a more general definition, psychophysiology can thus be defined as the science of understanding the link between psychology and physiology - the combined study of the mind and the body and their ability, consciously or subconsciously, to affect each other.

In the early phases of psychophysiology, the main focus of interest was on studying galvanic skin response (later to be known as electrodermal activity). As the years passed, the focus gradually shifted towards studying brain activity, much due to improved measuring devices and increased interest in cognitive functioning.

Studying stress and stress related disorders is closely linked to psychophysiology, and the dysfunctions examined range from anxiety attacks to burnout syndromes, sleeping disorders to post-war stress. Within this research, studying the autonomic nervous system (ANS) [34] is a crucial factor. The ANS consists of the sympathetic, the parasympathetic, and the enteric nervous system. The first two are of immediate interest to a diagnosing clinician. The sympathetic nervous system is the part of the ANS which is active during stressful situations. From a historic perspective, the necessity of such a system can be linked to situations such as fighting a tiger or running from a forest fire. The parasympathetic system on the other hand becomes active under relaxing conditions, such as during rest and during sleep. As a result of stress, the sympathetic system can become dysfunctional and trigger too easily or be constantly active, potentially resulting in a number of stress related symptoms and disorders. Psychophysiological stress research is thus interested in stabilizing the balance between the sympathetic and the parasympathetic parts of the nervous system.

One of the more commonly used treatment methods in psychophysiology is known as biofeedback [35, 36]. Biofeedback is focused on gradually learning to control one or more aspects of psychological or physiological factors. For stress treatment, a patient is often instructed to gradually learn how to control their breathing, while witnessing the immediate effects of their improved respiration technique (decreased or increased heart rate, increasing finger temperature etc) on a screen. Such a method is typically combined with regular physical exercises, learning how to eat properly, and general relaxation techniques.

Researchers in psychophysiology are equally interested in ways of studying, observing and recording measurable factors affecting the ANS. One of the most important factors is Respiratory Sinus Arrhythmia (RSA) [37, 38]. Respiratory Sinus Arrhythmia (RSA) is defined as the variation in heart rate (heart beats per minute) that accompanies breathing, known as Heart Rate Variability (HRV) [39]. During inhalation, the heart rate increases, and decreases again during exhalation. On a physiological level, the heart rate variation occurs as a result of different activity levels of the sympathetic and the parasympathetic parts of the autonomous nervous system during different stages of the respiration cycle Physicians studying causes and effects of psychophysiological disorders such as stress and stress related diseases have a valuable tool in the study of RSA and RSA patterns. Its usefulness lies primarily in the ability to help indicate irregular heart rate patterns, some of which may be caused by stress.

Computers come into practice in psychophysiology much the same way it does for most medicine: for analyzing, classifying, filtering, and storing measurements and patient data. Case-Based Reasoning has been used extensively in the medical domain. For an overview of medical systems utilizing CBR, as well as a description of general advantages and disadvantages of CBR in medicine, paper B in this thesis represents a good start.

3.2.2 Motivation

The majority of analysis of sensor output data in the medical domain is today performed manually, and this is true for psychophysiology as well. Although it is not always possible or even desirable to replace the physician with automated systems, repetitive parts of the analysis or classification can often be automated successfully, leaving the more complicated analysis and the final decision making to the physician. In the domain of stress research, analysis of RSA is a time-consuming, manual process. In recent work by Nilsson [40], single RSA respiration cycles have successfully been classified using CBR. However, to draw conclusions about a patient it is generally necessary to look at *sequences* of RSA respiration cycles. This process is currently done manually as well, and is harder to automate since there are no available cases available to insert into a case base - the analysis is mostly done on an intuitive level. The motivation is thus twofold: to automate another level of RSA analysis, and to help define clear-cut cases for an RSA sequence case library.

In Paper C, a method of clustering and analyzing sequences of previously classified RSA patterns is proposed. The method utilizes a combination of automated data analysis using clustering, and manual verification and construction of stereotypes by experts.

3.2.3 Related work

Computer-aided medical diagnose systems have been around since the 1970's, the first system being MYCIN, a system to diagnose blood infections [41]. These early systems were generally completely rule-based, and often successful in their domains, but sometimes suffered from maintenance and rule inconsistency problems [42].

Stereotypes have not been used in the domain of medical diagnosis. In fact, CBR is in itself a relatively new field of research within medical diagnose systems research. For examples of the usage of CBR in medical diagnosis, see [43, 44, 45, 40]. One of the earliest medical expert systems utilizing CBR was CASEY [46], that combines a CBR approach with a model based expert system for diagnosing cardiac diseases (heart failure).

Although there have been many attempts at creating fully functional diagnose systems, experts are typically still a crucial part of the decision chain at later stages. The principal, emerging value of computers in medicine over the past several decades has first and foremost been one of organizing and communicating information about patients [47].

There have been no reported attempts at creating a medical diagnosis system for the particular task of diagnosing patients with stress related diseases and/or symptoms. Two examples that deal with vaguely similar topics are Montani [48], who in her Ph.D. thesis explores the use of decision support in diabetes care, and Marling and Whitehouse [49], who examine the possibilities of using CBR for prescribing drugs to patients suffering from Alzheimer's disease.

3.2.4 Contribution

An iterative method for semi-automatic creation of case libraries in poorly understood domains. The relationships between previously classified single respiration cycle RSA are complex and not fully understood. Paper C presents an approach towards analyzing classified RSA sequences using a combination of clustering and manual generalization based on available expert knowledge.

Chapter 4

Paper Contributions

This thesis includes three papers, inserted in chronological order.

Paper A, *Category-Based Filtering and User Stereotype Cases to Reduce the Latency Problem in Recommender Systems*, was presented at the 6th European Conference on Case-Based Reasoning (ECCBR) 2002, held in Aberdeen, Scotland. ECCBR is held every even year, intertwined with the International Conference on Case-Based Reasoning (ICCBR). Paper B, *Advancements and Trends in Medical Case-Based Reasoning: An Overview of Systems and System Development*, was presented at the 17th International FLAIRS conference 2004, in Miami, Florida. The third paper, *Building a Case Base for Stress Diagnosis: An Analysis of Classified Respiratory Sinus Arrhythmia Sequences*, was presented at the Case-Based Reasoning in Medicine Workshop at the 7th ECCBR conference, held in Madrid, Spain, 2004.

4.1 Paper A

Paper A presents a novel approach to reduce the latency problem, which is often encountered by WWW Recommender systems. The filtering technique most often used in Recommender Systems - collaborative filtering - depends on implicit or explicit peer reviewing of items to enable finding users with similar preferences. The latency problem represents the difficulty of introducing new, unrated items into such a system. Since new items have not yet been rated by any user, they are not included in any recommendations, which leads to few users getting a chance to rate them, and thus, the problem persists. This paper presents an approach based on clustering users' category preferences (as opposed to item preferences) to build user stereotypes containing probabilistic relationships between categories of items. The user stereotypes represent cases and are used both to identify which users belong to a particular stereotype, and to recommend items - a combined approach referred to as category-based filtering. The latency problem is reduced since items can be recommended early on based on their category. An additional advantage of the offline clustering approach compared to online collaborative filtering is decreased response times.

Mikael Sollenborn is the main author of the paper. Peter Funk contributed with valuable ideas and discussions.

4.2 Paper B

Paper B is both a survey over influential CBR systems in medicine from 1998 and onwards, and an attempt to identify recent trends in the construction and usage of such systems. The description of each system is based around a number of properties described in the paper, such as adaptation, autonomicity, case base size, commercial usage etc. The paper also describes the general advantages and disadvantages of using CBR in medicine. Aside from descriptions of each system, the paper revolves around a questionnaire sent to the authors of the papers describing the systems included in the survey. Having the original authors comment on the properties of their own system ensures a high level of certainty about the correctness of the presented system property matrix. Among the more interesting trends are the increasing usage of hybrid systems, separate pre-processing, and, somewhat unfortunately, a relatively low level of commercialization and every-day usage of developed systems.

Mikael Sollenborn and Markus Nilsson contributed equally to the paper - authors are listed in alphabetical order. Mikael Sollenborn authored the introduction, the property descriptions, some of the system descriptions, and half of the trends section. Markus Nilsson authored the majority of system descriptions, the other half of the trends section, and conclusions.

4.3 Paper C

Building upon previous work on classifying Respiratory Sinus Arrhythmia (RSA), this paper presents an approach to analyze sequences of classified RSA patterns. Each RSA classification consists of one respiration cycle, from inhalation to exhalation, and provides valuable information for a physician looking at separate respiration cycles. However, a typical clinical measurement session with the system in use contains 60-80 respiration cycles, and analyzing the meaning of recurring sequence patterns enables RSA analysis on a higher level. The major difficulty in this process is building the case base. Unlike classifying single respiration cycles, there is little agreed upon knowledge about meaningful patterns of dysfunctions following upon each other, which effectively means that the case base has to be constructed out of existing session data rather than expert knowledge. In this paper, clustering is utilized to identify recurring RSA sequences, where separate phases of each session are first clustered and analyzed by physicians, and later reassembled to allow for clustering of complete sessions. In the next stage, physicians are again employed to manually identify valid stereotype cases among the recurring session patterns. The goal of this process is to create a case base that can be used as the second layer in a CBR-oriented RSA decision support system.

Mikael Sollenborn is the main author of the paper. Markus Nilsson contributed with valuable ideas and discussions.

Chapter 5

Conclusions and Future Work

5.1 Summary and conclusions

This thesis presents a hybrid approach towards handling problems related to grouping data, and utilizing the data in the form of user stereotypes. The two application domains are personalization for e-commerce systems on the WWW, and RSA sequence analysis in the psychophysiological domain.

The main contributions of this thesis are

- A novel approach to combining collaborative and content-based filtering for reducing the latency problem
- An iterative method for semi-automatic creation of case libraries in poorly understood domains.

5.2 Future work

Completing and fully evaluating the construction of a case library for sequences of classified RSA patterns. As previously mentioned, Paper C describes an approach to creating user stereotypes cases in order to construct a case library in an area where domain knowledge is not fully developed. A full empirical evaluation of this approach in a clinical environment would be extremely valuable.

Building a complete RSA classification system. A complete RSA classification system could be constructed by integrating the case library constructed using the approach described in Paper C with previously developed layers of RSA analysis and classification. The resulting single CBR system for computer-aided classification and diagnosis would hopefully have important consequences for the effectiveness and understanding of RSA analysis in the future.

Bibliography

[1] Ian Watson. *Applying Case-Based Reasoning: Techniques for Enterprise Systems*. Morgan Kaufmann Publishers, 1997.

[2] J.L. Kolodner. *Case-Based Reasoning*. San Francisco: Morgan Kaufmann Publishers, 1993.

[3] David B. Leake. *Case-based Reasoning: Experiences, Lessons, and Future Directions*. AAAI Press/MIT Press, Massachusetts Institute of Technology, Cambridge, Massachusetts 02142, 1996.

[4] Laurene Fausett. *Fundamentals of Neural Networks*. Prentice Hall, Upper Saddle River, New Jersey 07458, USA, 1994.

[5] Rainer Schmidt, Bernhard Pollwein, and Lothar Gierl. Experiences with case-based reasoning methods and prototypes for medical knowledge-based systems. In *Artificial Intelligence in Medicine*, pages 124–132. AIMDM'99, June 1999.

[6] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa. Improving the effectiveness of collaborative filtering on anonymous web usage data. In *Workshop Intelligent Techniques for Web Personalization, editors S. Anand, B. Mobasher*, pages 53–60. IJCAI'2001, 2001.

[7] Elaine Rich. User modeling via stereotypes. *Cognitive Science*, 3:329–354, 1979.

[8] G. Brajnik and C. Tasso. A shell for developing non-monotonic user modeling systems. *International Journal of Human-Computer Studies*, 40:31–62, 1994.

[9] J. Kay. The um toolkit for cooperative user modeling. *User Modeling and User Adapted Interaction*, 4:149–196, 1995.

[10] G. Paliouras, V. Karkaletsis, C. Papatheodorou, and C.D. Spyropoulos. Exploiting learning techniques for the acquisition of user stereotypes and communities. In *Proceedings of the International Conference on User Modelling*. UM'99, 1999.

[11] C. Moghrabi and M.S. Eid. Modeling users through an expert system and a neural network. *Computers and Industrial Engineering*, 35(3-4):583–586, December 1998.

[12] A.K. Jain and R.C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.

[13] A.K. Jain, M.N. Murty, and P.J. Flynn. Data clustering: A review. *ACM Computing Surveys (CSUR)*, 31(3):264–323, 1999.

[14] A.R. Webb. *Statistical Pattern Recognition*. John Wiley and Sons Ltd, 2002.

[15] L. Hubert. Approximate evaluation techniques for the single-link and complete-link hierarchical clustering procedures. *Journal of the American Statistical Association*, 69:698–704, 1974.

[16] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297, 1967.

[17] T. Kohonen. *Self-Organizing Maps*. Springer Series in Information Sciences, Vol. 30, 1995.

[18] Editor D. Riecken. *Communications of the ACM 43. Special Issue on Personalization*, 8, 2000.

[19] M.D. Mulvenna, S.S. Anand, and A.G. Buchner. Personalization on the net using web mining. *Communications of the ACM 43*, 8:123–125, 2000.

[20] P. Hagen. Smart personalization. *The Forrester Report*. Forrester Research, Cambridge, Massachusetts, 1999.

[21] P. Cotter and B. Smyth. Personalisation technologies for the digital tv world. In *Prestigious Applications of Intelligent Systems*. PAIS'00, 2000.

[22] D.W. Oard and G. Marchionini. A conceptual framework for text filtering. University of Maryland, College Park, MD 20742, 1996.

[23] M. Pazzani, J. Muramatsu, and D. Billsus. Syskill & webert: Identifying interesting web sites. In *Proceedings of the ACM SIGIR'96 Conference*. SIGIR'96, 1996.

[24] Elaine Rich. Stereotypes and user modeling. *User Models in Dialog Systems*, pages 31–51, 1989.

[25] A. Jameson. Generalizing the double-stereotype approach: A psychological perspective. In *UM92: Third International Workshop on User Modelling*, pages 69–83. UM'92, 1992.

[26] D.N. Chin. Knome: Modelling what the user knows in uc. *User Models in Dialog Systems*, pages 133–162, 1989.

[27] Matthew N. Dailey, Gregory S. Miller, and James C. Lester. Exploiting stereotypes to eliminate strategic bias. In *Proceedings of the 5th International Conference on User Modeling*. UM'96, January 1996.

[28] L. Ardissono and D. Sestero. Using dynamic user models in the recognition of the plans of the user. *User Modeling and User-Adapted Interaction*, 5(2):157–190, 1996.

[29] T. Kuflik, B. Shapira, and P. Shoval. Stereotype-based vs. personal-based filtering rules in information filtering systems. *Journal of American Society of Information Systems Technology (JASIST)*, 2003.

[30] Nicola Henze and Wolfgang Nejdl. Logically characterizing adaptive educational hypermedia systems. In *Proceedings of the AH'2003*. AH'2003, 2003.

[31] C. Hayes, P. Cunningham, and B. Smyth. A case-based reasoning view of automated collaborative filtering. In *Proceedings of 4th International Conference on Case-Based Reasoning*, pages 243–248. ICCBR'01, 2001.

[32] K. Funakoshi and T. Ohguro. A content-based collaborative recommender system with detailed use of evaluations. In *Proceedings of 4th International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies*, pages 253–256. KES'00, 2000.

[33] J.A. Stern. Towards a definition of psychophysiology. *Psychophysiology*, 1:90–91, 1964.

[34] C.R. Noback and R.J. Demarest. *The Nervous System, Introduction and Review.* McCraw Hill, 1986.

[35] C.F. Weems. The evaluation of heart rate biofeedback using a multielement design. *Journal of Behaviour Therapy and Experimental Psychiatry*, 29:157–162, 1998.

[36] Marie J.Cowan, Helen Kogan, Robert Burr, Sue Hendershot, and Lynne Buchanan. Power spectral analysis of heart rate variability after biofeedback training. *Journal of Electrocardiology*, 23:85–94, 1990.

[37] J.A. Hirsch and B. Bishop. Respiratory sinus arrhythmia in humans: How breathing pattern modulates heart rate. *American Journal of Physiology*, 241:620–629, 1981.

[38] Tuan Pham Dinh, Helene Perrault, Pascale Calabrese, Andre Eberhard, and Gila Benchetrit. New statical method for detection and quantification of respiratory sinus arrhytmia. *IEEE Transactions on Biomedical Engineering*, 46(9):1161–1165, September 1999.

[39] Marek Malik and John Camm. Heart rate variability -standards of measurement,physiological interpretation,and clinical use. *European Heart Journal*, 17:354–381, March 1996.

[40] Markus Nilsson and Peter Funk. A case-based classification of respiratory sinus arrhythmia. In *Proceedings of the The Seventh European Conference on Case-Based Reasoning.* ECCBR'04, 2004.

[41] B.G. Buchanan and E. H. Shortliffe. *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project.* Addison-Wesley, 1984.

[42] William E. Spangler and Jerrold H. May. Success and failure in cooperative expert systems development: a tale of two projects. *Journal of Systems and Software archive. Special issue on expert systems that failed*, 19(2):131–140, 1992.

[43] C. Bradburn and J. Zeleznikow. The application of case-based reasoning to the tasks of health care planning. In *Proceedings of European Workshop on CBR*, pages 365–378. EWCBR'93, 1993.

[44] Rainer Schmidt and Lothar Gierl. Case-based reasoning for medical knowledge-based systems. In *German Workshop on Experience Management*, pages 720–725. Proceedings of MIE'00 and GMDS'00, 2000.

[45] Lothar Gierl and Rainer Schmidt. Cbr in medicine. *Case-Based Reasoning Technology, from Foundations to Applications*, pages 273–297, 1998.

[46] Phyllis A. Koton. *Using Experience in Learning and Problem Solving*. MIT Press, 1988.

[47] H.J. Murff and J. Kannry. Physician satisfaction with two order entry systems. *Journal of the American Medical Informatics Association*, 8(5):499–509, 2001.

[48] S. Montani. *Knowledge Management and Decision Support in Diabetes Care through Multi Modal Reasoning. Ph.D. thesis*. University of Pavia, 2001.

[49] Cindy Marling and Peter Whitehouse. Case-based reasoning in the care of alzheimer's disease patients. In *Case-Based Research and Development*, pages 702–715. ICCBR'01, July/August 2001.

II

Included Papers

Chapter 6

Paper A: Category-Based Filtering and User Stereotype Cases to Reduce the Latency Problem in Recommender Systems

Mikael Sollenborn, Peter Funk
In Proceedings of the 6th European Conference on Case-Based Reasoning, Aberdeen, September 2002

Abstract

Collaborative filtering is an often successful method for personalized item selection in Recommender systems. However, in domains where items are frequently added, collaborative filtering encounters the latency problem. Characterized by the system's inability to select recently added items, the latency problem appears because new items in a collaborative filtering system must be reviewed before they can be recommended. Content-based filtering may help to counteract this problem, but runs the risk of only recommending items almost identical to the ones the user has appreciated before. In this paper, a combination of category-based filtering and user stereotype cases is proposed as a novel approach to reduce the latency problem. Category-based filtering puts emphasis on categories as meta-data to enable quicker personalization. User stereotype cases, identified by clustering similar users, are utilized to decrease response times and improve the accuracy of recommendations when user information is incomplete.

6.1 Introduction

Personalization on the Internet is today a growing research area, as the information overload problem has created an emerging need for individualized user treatment. By focusing on each visitor's requirements, the user's effort in navigating vast amounts of information can be made more focused, efficient and manageableze. The underlying idea of personalization is the assumption that individualized content will satisfy users and increase revenue directly or indirectly, e.g. attract new users and make them more willing to revisit a web site and buy more services and products [1].

For personalization of web pages, Recommender systems are currently the most common approach. Based on the information filtering technique known as automated collaborative filtering (ACF) [2, 3, 4], standard Recommender systems essentially function on a peer review basis. When making recommendations, users with similar preferences are identified, and their item ratings are used to propose items to one another. Implementation of an ACF Recommender system can be divided into three steps [5]:

1. Record the behavior of a large number of people, e.g. their interest in selected items such as adverts, news, books, etc.
2. Select a number of users whose past behavior is similar to the current user.
3. Recommend personalized items based on preferences of the selected users.

In addition to collaborative filtering, personalized selections based on matching the current user's previous selections with individual items - known as content-based filtering - is also very common [6]. In short, where filtering with ACF involves comparing a user with other users, content-based filtering is performed by comparing the user's preferences with the available information about items, e.g. meta-data or content keywords.

One potential problem with standard Recommender systems is that all reasoning is done online. With impatient users waiting for quick responses, the search for similar users must be very time-efficient. This time restriction also results in fewer possibilities when trying to improve or extend the content filtering strategies. In order to improve both speed and recommendation effectiveness, current approaches to building Recommender systems often try to perform some of the reasoning offline using clustering techniques [7, 8].

Traditional Recommender systems also encounter the *latency* problem [9], i.e. new items incorporated into a Recommender system cannot be used in collaborative recommendations before a substantial amount of users have evaluated it, as the recommendations rely on other users opinions. This problem is especially apparent in domains where new items are often added and old items quickly get out of date. Content-based filtering may be a solution, but runs the risk of only recommending items almost identical to the ones the user has appreciated before [9]. As noted in [10], the most obvious solution to the latency problem is to categorize the items in the system. In this paper we go one step further and assume that for some applications domains, Recommender systems solely based on categories provide sufficient personalization.

Our proposed approach for reducing the latency problem in highly dynamic domains is called category-based filtering. In a category-based filtering system, user preferences reflect attitudes not towards single items, but categories of similar items, both on a collective and an individual level. At the collective level, off-line clustering is used to find user stereotype cases, thus employing a Case-Based Reasoning view of information filtering. Clustered user data enables quicker response times and makes collaborative reasoning possible for meta-data in the form of categories. In section 3.2 the category-based filtering approach is explained. Section 3.3 gives a more detailed exploration of classification, clustering and item selection. The research prototype, a personalization system based on category-based filtering and user clustering, is briefly described in section 3.4, and the following section gives a conclusion.

6.2 Category-based filtering approach

In this section we explore how category-based filtering is used in a Recommender system and how it is integrated with clustering and user modeling.

6.2.1 Rating technique

Typically, rating methods are divided into invasive and non-invasive techniques. An invasive rating method requires explicit user feedback. A commonly used approach is to let users mark their appreciation of items viewed or purchased on a scale. In contrast, non-invasive methods observes the user's behavior, requiring no more input than the user's normal interaction with the system. As a result, non-invasive methods generate noisier data, but have the benefit of being invisible to the user. For our purposes, i.e. dynamic domains where data changes frequently, invasive techniques put too much burden on the users. Instead, a simple non-invasive technique was chosen. The system selects a set of items to show, and observes user reactions to these items. The system notes whether the user responds positively, by clicking on one of the currently shown items, or negatively, by ignoring them. We do not consider viewing time following a click, because the number of responses to a category of items will be many times as high as that for single items in a representationless ACF system, making the consequences of a single click less relevant.

6.2.2 Category-based filtering

We refer to the personalization approach proposed in this paper as *category-based filtering*. Its main characteristic is that selection of information is based on category ratings instead of item ratings, in contrast to other content-filtering strategies in general, and representationless collaborative filtering in particular. To function, category-based filtering requires categorization of every item, either manually or by an automated process.

In our implementation of category-based filtering, the selection of items is based partly on individual user models, and partly on collective user stereotypes cases. A *user model* represents the current knowledge about a user's reaction towards shown categories of items. A *user stereotype case*, in contrast, consists of collective information about a group of users.

User stereotypes, as introduced by Rich in [11], require two types of information. The system must know what properties capture a stereotype, and what events or behavior that implies a particular stereotype. On the Internet, this information is highly dynamic and in our domain dependent on both content of categories and population of users. A clustering approach, as described below, is therefore preferable to static stereotypes, since it is able to automatically identify related categories and adapt to a changing population of users and their preferences.

By representing a solution to the problem of supplying a 'typical' kind of user with appropriate information, it is natural to see user stereotype cases as part of a Case-Based Reasoning process. When the information in a user model is insufficient for deciding which items to select, the user stereotype case most closely resembling the user is consulted to make assumptions about the user's expected behavior (Retrieve, Reuse). The case is Revised when the user evaluates the recommended items, and Retained when the user stereotype cases are updated.

The system could also be seen as a hybrid of collaborative and content-based filtering, with strong emphasis on categories as item meta-data. Unlike other such hybrid systems [12, 4], the collaborative selection is also based on meta-data, as the peer reviewing process deals with categories instead of items.

The focus on categories reduces the latency problem, as new items can be recommended as soon as the system knows the user's attitude towards the item's corresponding category. Because of this, selecting items based on category ratings instead of ratings of individual items is especially suited for domains where there is a constant flow of new information e.g. news and adverts), provided that effective categorization is possible.

As category-based filtering could possibly be seen mainly as an extension to existing filtering strategies, one might feel inclined to propose the terms category-based collaborative filtering and category-based content-based filtering instead. However, apart from the clumsiness of these expressions, the term category-based collaborative filtering has been used for other purposes [13].

The user stereotype cases needed for collaborative selection of items are created offline using clustering methods described in section 3.3.

Each cluster represents a part of the entire user population. Probabilistic nets within the cases, formed from the cluster data, represent collective attitudes towards categories of items.

With the clusters identified, a new user can be assigned a user stereotype case after a short period of initial observation. As the user model matures, the case assignment may change to point out the characteristics of the user in a more precise way.

The frequency of generating clusters and updating the user stereotypes cases depends on the application domain, e.g. the number of visits to the web site it's being used on. All individual user information is always preserved, enabling the system to perform a re-clustering at any time.

User models

Based on the dimensions identified in [11], a user model in the proposed system has the following properties: each user has a separate user model, the model is built and refined non-invasively by the system on each site visit, and the model contains both specific, short-term information and (limited) long-term information.

A user model is represented by a matrix of choices and preferences For each category, the number of times the user has been approached with items belonging to it is stored, as well as the number of positive responses. Figure 6.2.2 shows an example preference matrix with a simplified history ("Last ten"-column) that would capture sudden changes of user preferences. In this example, only two clicks the last ten times the user was approached with hunting items may indicate a decline of interest for such adverts.

When the preferences of a user are to be ranked, the value for each category may be reduced, e.g. to one of four levels: positive, neutral, negative, or unknown. The unknown attitude is reserved for categories that have not yet been evaluated.

	Shown	Clicked	Last ten
Hunting	24	11	2
Fishing	18	4	3
Cosmetics	12	1	0

Figure 6.1: Example preference matrix.

User stereotypes and appreciation nets

User stereotypes cases are representations of common attitudes among a group of similar users. The chosen method of capturing collective interests is to utilize what will be referred to as appreciation nets. Appreciation nets are graphs with nodes and directed edges, where edges represent a probabilistic relationship. If every node in the net has an edge going into every other node, the appreciation net is said to be complete, with n(n-1) node connections. In Figure 6.2 an example of an appreciation net is given for four item categories. In this example population the likelihood that a person who likes hunting is also interested in motor sports is 60% (indicated by 0,6 at the edge from hunting to motor sports). In the opposite direction, a person that appreciates motor sports also enjoys hunting with a probability of 30%. Of all the persons belonging to this population, 50% enjoy motor sports, but only 20% appreciates hunting, as indicated in the category nodes.

6.2.3 System architecture

Figure 6.3 shows a schematic view of a system using category-based filtering. Each user visiting the web site is assigned an individual user agent. The agent's task is to handle web page modifications and interaction with the user involving personalized items. The Reasoner uses category-based filtering to select a set of items assumed to be of interest for the current user, based on the user model (the user's preference matrix) and the closest user stereotype case. The user agent tracks all user responses and stores them in the user model. The cases are updated offline (as indicated by the dotted line) by clustering similar user models.

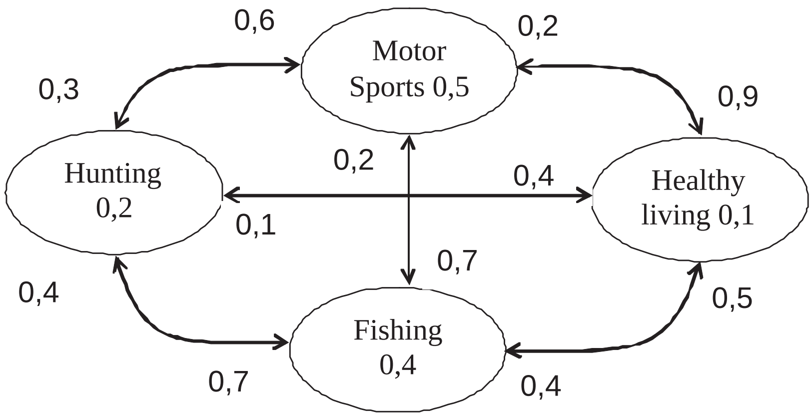

Figure 6.2: An appreciation net with four item categories.

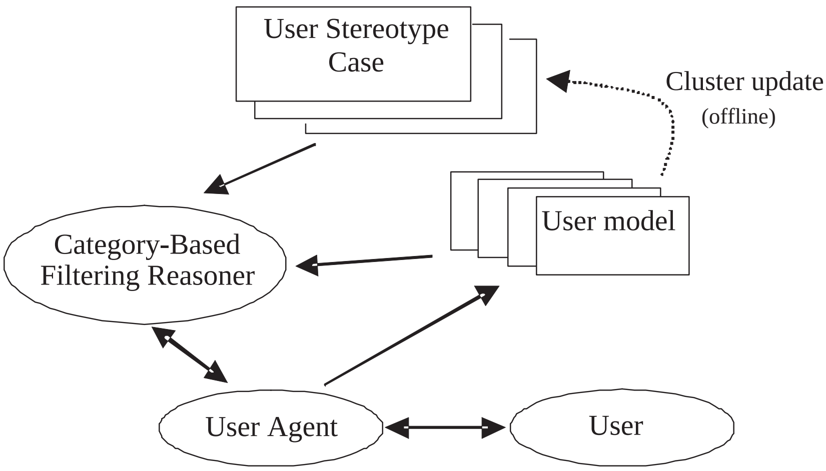

Figure 6.3: Example of a personalization system using category-based filtering.

6.3 Clustering and selections

This section takes a closer look at techniques and algorithms used for personalization in a category-based system.

6.3.1 Clustering users

For the creation of user stereotype cases, an agglomerative, hierarchical clustering method was chosen, avoiding partitioning since the number of appropriate clusters will be difficult to guess in advance. The variables determining cluster membership are as many as there are categories in the system, but categories that have not yet been evaluated by a user are not included when comparing him/her to other users. Different values are assigned to the category attitudes negative, neutral, and positive. To measure distance between clusters, the city-block (Manhattan distance) metric is used. Similar clusters are merged using the unweighted pair-group average method [14].

After clustering, the (highly subjective) optimal number of clusters must be determined. Currently, the chosen method is to pick a maximum number of clusters M based on the number of users N registered on the web site. The minimum amount of users U belonging to a cluster is then calculated as $U = N/M$, meaning we treat a group of U or more similar individuals as "statistically significant" to form a collective model. Now, traversing the cluster tree selecting clusters of size U or bigger results in a number of clusters from 1 to M. Although this method produces acceptable results for the limited test domain, more formal approaches for determining the number of clusters, such as MDL or BIC, are considered for the final implementation.

As noted earlier, a user stereotype case contains an appreciation net, with all nodes connected to every other node in both directions. When forming such a net, a joint distribution is made from the ranked category preferences of every user belonging to the group, resulting in a two-dimensional matrix exposing collective preferences. Building the user stereotype case is primarily a question of how much information from the joint distribution to include in the appreciation net.

For each category C the system stores the probability of a positive evaluation by any user belonging to the group, as node values in the appreciation net. This information is very important because for categories where $P(C)$ is high, quick tests can be made to determine whether new users conform to a specific cluster.

Secondly, couplings between pairs of categories are examined. The probability of a user appreciating C in case the user likes D, $P(C|D)$, is preserved for each category-to-category connection, stored as binary relationships between category nodes in the appreciation net.

The dominating preference information can be captured in binary relations between category nodes. Some preference information may also be captured in probability values involving more than two category nodes e.g. if users are interested in category D and E the likelihood for interest in C is $P(C|D \wedge E)$). A calculation of probability values among all possible n-tuples of relations may have a too high computational price and make the resulting model unnecessarily complex. Considering the possibly increased inference ability gained from probability relations involving three of four categories, such probability values may be worth preserving in the appreciation net if the values are distinctive enough, i.e. exceptionally low or high).

6.3.2 Classification of users

Automatic classification is attempted by targeting the user with information corresponding to differential probability values in the user stereotype cases appreciation nets. The goal is to determine which case resembles the new user the most.

Beginning with the biggest cluster is an adequate starting point, because it's where the user most likely belongs. What is sought for is a number of category nodes in the appreciation net with high appreciation probabilities, with these values being as unique as possible compared to the equivalent category values in other cluster models. Categories with unique but low appreciation probabilities are not as interesting, as a positive response can be considered a lot more informational than a negative, i.e. a user showing interest probably is interested, but a user that does not may simply be temporary ignorant or short of time. The appropriateness of being chosen is calculated for every category node C by comparing it to the corresponding category node C_i in every other case appreciation net, using

$$F = P(C) \times \sum_{i=1}^{n} |P(C) - P(C_i)| \tag{6.1}$$

The three categories with highest F are chosen for testing, meaning information belonging to these categories will be shown to the user, with five items per category. This may or may not take several visits to the site, depending on the type of site and how much information can be shown normally during a visit. When all items have been shown, an initial user stereotype case membership determination is performed. The formula resembles a Naive Bayesian Classifier, but sums probabilities instead of multiplying them to avoid having occasional conditional probabilities close to zero produce an unsuitably low total similarity value. The categories involved in the test are compared to each user stereotype case, putting emphasis on similar categories with high probability values (again because it's more important what is appreciated than what is not), and limiting the comparison to categories already tested. The similarity S is calculated for every user stereotype case, where C_i is category i in the tested user's preference matrix, C_{sui} the corresponding category i for member u in user stereotype s, R an empirically chosen similarity range [0,1], and M an empirically chosen modification rate [0,1], using

$$S = \sum_{i=1}^{n} P(P(C_{sui}) = P(C_i) \pm R|s) \times (P(C_{si}) \times M + (1 - M)) \tag{6.2}$$

The case that most closely resembles the initial behavior of the user (highest S) is now chosen for a second pick of categories using equation (6.1), separate from the ones chosen before. A new test is done, followed by another comparison using equation (6.2). This process continues every time the user visits the site. Eventually, the system will mix the testing data with items selected by assuming that the user does in fact belong to the cluster the user currently resembles the most, as well as re-evaluating categories that were positively responded to before. As more and more categories are evaluated, the amount of testing data ceases gradually. Cluster membership may still change, either because the number of clusters or the user's behavior has changed, but no longer as a result of evaluating "classification-aggressive" testing data.

6.3.3 Selection of personalized items

Once a user is assigned to a user stereotype case, the Reasoner (figure 6.3) is able to make qualified guesses about what a semi-new user might and might not appreciate. Whenever there is insufficient information about a user as an individual during decision-making, the case connected to the user will be examined to find out how similar users have behaved.

Asked to select personalized information for a specific user, the system initially decides whether or not it knows enough about the user's behavioral patterns to determine cluster membership. If not, the system will try to classify the user as described above. If cluster affiliation can be guessed but not completely determined, the system may alternately pick items it assumes the user will appreciate, while at the same time trying to further strengthen the belief that the user belongs to a specific cluster.

The Reasoner selects two types of information, appreciation-known and appreciation-assumed. A new user is confronted with a lot of appreciation-assumed information, but as the user provides more information, the appreciation-known information gradually replaces it.

Appreciation-known items are chosen only from categories the user has 'sufficiently evaluated'. A sufficiently evaluated category simply means a category that the user has evaluated enough times to be reasonably sure about the individual's attitude towards it. The number labeled sufficient varies however, as the system gradually tries to follow a new or semi-new user when providing more information. The system may also decide that a category needs re-evaluation if the user's last ten responses to it has been significantly different to the corresponding long-term behavior.

Appreciation-known information is selected by ranking the user preferences, picking items from categories that have been positively evaluated. In this process, the system tries to balance the number of shown items among the positive preferences, as well as sometimes picking sufficiently evaluated categories with a less positive ranking to allow for re-evaluation. When appreciation-assumed items are to be selected, the system chooses a category node starting point in the appreciation net among the users positively ranked preferences. With this node as base, the system examines all connected category nodes. The category to select information from is chosen randomly from a dynamically generated pie chart, where each category not among the user's positive preferences gets a slice size (choice probability) calculated using equation (6.3). W is the connection weight, C is the number of clicks done on items belonging to this category, S the number of times shown to the user, and L how many of the last H items in the category that has been clicked by the user. H is domain dependent; in our test evaluation the history length is ten items, as shown in Figure 6.2.2.

$$P = W \times ((C+1)/(S+1) + (L+1)/H) \tag{6.3}$$

Another form of appreciation-assumed item selection, used in parallel with the method above, essentially uses the same method as the automatic classification process: picking items from categories in the appreciation net where the probability of a positive response is high. This item selection method is used only if there are still categories with high appreciation probabilities that the user has not yet evaluated. The items selected by using each of these techniques are finally merged, and presented to the current user.

6.4 Implementation

The research prototype is currently being implemented, and has so far been tested on a small set of users in a limited surrounding. News and adverts were chosen as item types, as they both represent dynamic domains were items generally change often. In the testing environment, users are shown a selected number of news summaries, containing approximately 200 letters. These news items are categorized manually in advance. The user is able to receive the full article by clicking on a news item. This information is used to build a preference matrix for the current user to aid in the category-based filtering approach. Adverts are handled similarly. A selected number of adverts are shown to the user and when clicked, additional product information is displayed.

To keep the user models reasonably small, item selections in categories are not given any time stamps, only the number of positive responses for the last ten times the item was shown (see example in Figure 6.2.2). A number of user stereotype cases have been initiated in advance, and offline update of clustering is performed frequently in the form of refinement (a new cluster is generated from the same set of users). The re-clustering algorithm for grouping similar users into new clusters, as discussed in section 3.2, has been implemented but remains to be thoroughly tested and integrated. The appreciation nets currently only capture binary relations between item categories.

The prototype has been evaluated by a number of hypothetical users. The testing of the prototype has shown ability to quickly adapt to users preferences for a small number of categorized news and adverts (50 news and 70 adverts, 5 different hypothetical users with their interest profile predetermined and with a consistent behavior and low amounts of noise).

6.5 Conclusions

In this paper we have presented an approach to Recommender systems for application domains where items are frequently added. Provided that sufficient categorization is possible, we have shown that category-based filtering enables handling the latency problem.

In the proposed approach, users are represented partly by individual user models, and partly by user stereotypes cases. The cases, which are created offline through clustering, are used when the knowledge about an individual user is too limited to draw the needed conclusions for recommending items. The system will automatically attempt classification of new users by comparing the user's behavior with the user stereotype cases, selecting the most similar one.

Personalized information is divided into two categories: appreciation-known and appreciation-assumed. While the former represents item selections based on a user's known previous behavior, appreciation-assumed items are chosen because of high appreciation probabilities among other users belonging to the same user stereotype case as the current user. So far, category-based filtering has been tested on hypothetical users in a limited surrounding, where the approach has shown the ability to adapt to user's needs.

Large-scale tests to further confirm the usability of category-based filtering for practical domains are currently being prepared.

Bibliography

[1] F.F. Reichheld and W.W. Sasser. Zero defections : quality comes to service. *Harvard Business Review*, 68:105–111, September-October 1990.

[2] U. Shardanad and P. Maes. Social information filtering: Algorithms for automating 'word of mouth'. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 210–217. CHI'95, 1995.

[3] A. Konstan J, B.N. Miller, D. Maltz, J.L. Herlocker, L.R. Gordan, and J. Riedl. Grouplens: Applying collaborative filtering to usenet news. *Communications of the ACM 40*, 3:77–87, 1997.

[4] P. Cotter and B. Smyth. Waping the web: Content personalisation for wap-enabled devices. In *Proceedings of the 1st International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, pages 98–108. AH'00, 2000.

[5] R. Mukherjee and S. Dutta. Movies2go - a new approach to online movie recommendations. In *Workshop on Intelligent Techniques for Web Personalization*, pages 26–34. IJCAI'2001, Seattle Washington, 2001.

[6] R. Rafter and B. Smyth. Passive profiling from server logs in an online recruitment environment. In *Workshop on Intelligent Techniques for Web Personalization*, pages 35–41. IJCAI'2001, Seattle Washington, 2001.

[7] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa. Improving the effectiveness of collaborative filtering on anonymous web usage data.

In *Workshop Intelligent Techniques for Web Personalization, editors S. Anand, B. Mobasher*, pages 53–60. IJCAI'2001, 2001.

[8] L.H. Ungar and D.P. Foster. Clustering methods for collaborative filtering. In *AAAI Workshop on Recommendation Systems.* AAAI Press, Menlo Park California, 1998.

[9] K. Funakoshi and T. Ohguro. A content-based collaborative recommender system with detailed use of evaluations. In *Proceedings of 4th International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies*, pages 253–256. KES'00, 2000.

[10] C. Hayes, P. Cunningham, and B. Smyth. A case-based reasoning view of automated collaborative filtering. In *Proceedings of 4th International Conference on Case-Based Reasoning*, pages 243–248. ICCBR'01, 2001.

[11] Elaine Rich. User modeling via stereotypes. *Cognitive Science*, 3:329–354, 1979.

[12] M. Balabanovic and Y. Shoham. Combining content-based and collaborative recommendation. *Communications of the ACM 40*, 3:66–72, 1997.

[13] A. Kohrs and B. Merialdo. Using category-based collaborative filtering in the active webmuseum. In *Proceedings of IEEE International Conference on Multimedia and Expo*, volume 1, pages 351–354. ICME'00, 2000.

[14] A.K. Jain and R.C. Dubes. *Algorithms for Clustering Data.* Prentice Hall, 1988.

Chapter 7

Paper B: Advancements and Trends in Medical Case-Based Reasoning: An Overview of Systems and System Development

Markus Nilsson, Mikael Sollenborn
In Proceedings of the 17th International FLAIRS Conference, Special Track on Case-Based Reasoning, Miami, May 2004

Abstract

Case-Based Reasoning (CBR) is a recognised and well established method for building medical systems. In this paper, we identify strengths and weaknesses of CBR in medicine. System properties, divided into construction-oriented and purpose-oriented, are used as the basis for a survey of recent publications and research projects. The survey is used to find current trends in present medical CBR research.

7.1 Introduction

Ever since Shortliffe's seminal work on diagnosis of infection diseases [1], Artificial Intelligence has been applied in numerous applications in the health science domain. In the late 1980's, followed by ground-laying work done by Koton [2], and Bareiss [3], Case-Based Reasoning (CBR) appeared as an interesting alternative for building medical AI applications, and has since been further established in the field. Certainly, one of the intuitively attractive features of CBR in medicine is that the concepts of *patient* and *disease* lends itself naturally to a case representation. Although several advantages of using CBR in medicine has been identified, the medical field certainly is not without its problems, some of them specifically affecting CBR systems.

Gierl and Schmidt [4] identify the following key advantages of medical CBR;

- Cognitive Adequateness. CBR resembles the way physicians are reasoning about patients and the way they use their case expertise.
- Explicit Experience. A CBR system is naturally suited for adjusting itself to the specific requirements of a certain clinic or a surgeon.
- Duality of Objective and Subjective Knowledge. Instead of using the subjective knowledge of one or more experts to build systems (as is done for e.g. rule-based expert systems) CBR systems are built upon existing cases (which may or may not be fully understood).
- Automatic Acquisition of Subjective Knowledge. CBR systems exhibit an incremental knowledge acquisition, and knowledge can be abstracted by generalizing cases.
- System Integration. Patient records are already being collected by hospitals and practitioners and stored on machine readable mediums, which simplifies integration with CBR systems which can utilize them (after varying degrees of modification).

However, a number of disadvantages and problems can also be identified;

- Adaptation. Because of the often extremely large number of features involved in a medical case, adaptation of cases becomes problematic. Generalization and efficient feature identification methods helps to partly remedy this issue, but generally the problem persists. [5]

- Unreliability. Although the reliability of a CBR system increases with the proportion of coverage of the problem domain, reliability cannot be guaranteed. Adding new cases will not necessarily make a system converge towards greater reliability, as cases add only local improvement. Indeed, Bichindaritz argues that the strictly local properties of cases makes convergence an inappropriate notion for CBR systems.[6]

- Concentration on reference. CBR systems are concentrated on reference as opposed to underlying diagnostic factors. Thus, systems cannot function as sources of previous experience unless a suitable case exists in the case-base [7].

In this paper, we take a look at a number of the most influential medical CBR research projects in late years, with the aim of identifying trends in the development of such systems. Basing our work on the 1998 survey by Gierl and Schmidt [4], we focus primarily on systems created or reported about after 1998. In particular, we are interested in investigating if, and to what degree, the focus has changed on what type of medical CBR systems are constructed, and how they are constructed.

The method of identifying current trends involves examining systems from recent years by focusing on a set of distinctive system properties. We divide system properties into *purpose-oriented* and *construction-oriented*, where the first are characterized by the general type of action the system is supposed to perform (classification, planning, diagnosis, and tutoring), and the second indicate different types of constructions, such as systems supporting adaptation, hybrid systems, varying degrees of autonomicity etc. Additionally, we attempt to find trends of more general importance, looking at medical CBR systems from a broader perspective.

The rest of the paper is organized as follows. Next section contains a description of the different comparison properties used to differentiate a system from another. The section *Recent medical CBR systems* describes a selected number of influential works in the medical CBR domain. In section *Trends in medical CBR*, we present a system property matrix and identify construction-oriented and overall trends.

7.2 System properties

7.2.1 Purpose-oriented properties

With purpose-oriented properties, we refer to the separation of overall system purpose into planning, classification, tutoring, and diagnostic.

Diagnostic systems The majority of medical CBR systems belong in the diagnostic systems category. Diagnostic systems attempt to provide the user of the system with various degrees of assistance in the diagnosing process of a medical condition, possibly up to the point of a completely autonomous diagnose.

Classification systems Classification systems attempt to identify the group or group affiliations of real-world cases. One typical example is image classification systems that do not attempt a complete diagnosis.

Tutoring systems A medical tutoring system based on CBR is typically built closely around the concept of learning by examples, providing students with access to real patient cases.

Planning systems Planning systems are characterized by their intention to help in solving a process involving a number of steps. Therapy support is an often seen example of planning in medical systems.

7.2.2 Construction-oriented properties

Looking at medical CBR systems, we are interested not only in which systems have been recently constructed, but also how they were constructed and the motivation behind their construction. Once again, this is done to ease the identification of current trends in the development of medical CBR. However, in some cases it is not possible to derive the state of all these properties from the papers describing the projects in question.

Hybrid systems A hybrid medical CBR system denotes a multi-faceted solution to a problem space, using CBR as one of a number of AI technologies forming a complete system. Many such systems use CBR as the main organizer of data, and data-intense techniques such as neural networks to handle lower-level case identifications. Others match CBR with the Rule-Based Reasoning used in traditional expert systems to gain the advantages of both Rule-Based and Case-Based Reasoning.

Adaptive systems The problem of doing successful adaptation in the medical domain, because of the often enormous amount of features in a case, has been documented by Schmidt and Gierl [5]. In the system summary in section *Recent medical CBR systems*, we investigate if and to what degree medical CBR systems from recent years has started to utilize adaptation methods.

Case library size The size of the case library does not only involve the actual number of cases in the case library, but also the degree of case generalization into prototypes, i.e., the degree to which the system tries to merge existing cases into more general ones.

Autonomicity The degree of autonomicity is arguably of the most importance for diagnostic systems, where it denotes the level of interaction needed with a physician or corresponding medical expert before and after the diagnosis is complete. A purely autonomous system would produce diagnoses that would be accepted and used without having a human expert look at them, which is rarely the case in current systems. The degree of autonomicity implies the need for human intervention in the reasoning cycle and for evaluating its results.

Constraints System constraints concerns reliability and safety-criticality. Safety-criticality denotes the need to always provide correct answers, e.g., whether incorrect system behavior could potentially create dangerous or even life threatening situations. A system is reliable if it is always operational when needed.

7.3 Recent medical CBR systems

As was mentioned in the introduction, the focus of the survey is on systems created or reported about during the last five years. An overview of medical CBR systems before 1998 was done by Gierl et al. in [4]. From this overview, we adopted the division of systems into diagnostic, classification, tutoring, and planning systems.

7.3.1 Diagnostic systems

FM-Ultranet [8, 9] is a medical CBR project implemented with CBR-Works. FM-Ultranet detects malformations and abnormalities of foetus through ultrasonographical examinations. The detection, or diagnosis, uses attributes derived from scans of the mother's uterus, and identifies abnormal organs and extremities. Cases are arranged in a hierarchical and object oriented structure. The hierarchy is organized in 39 concepts, and every concept has one or more attributes. The attributes consists of anatomical features, medical history and general domain knowledge. Similarity between attributes in the concepts (objects) are mathematically calculated or compared through a look up table, depending on the attribute type. A report of the system's findings are generated when the detection (CBR) process is completed.

Perner [10] proposes a system that uses CBR to optimize image segmentation at the low level unit according to changing image acquisition conditions and image quality. The system has been used to detect degenerative brain disease in particular Alzheimer disease in CT images of a patient. The cases are comprised of images and image features as well as non-image information about the image acquisition and the patient. The solution of a case is the parameters of the image segmentation unit. The control of the parameter of the image segmentation unit is done by the CBR mechanism. This ensure high image quality of the output image. Similarity is calculated over the image information according to a special image similarity measure and over the non-image information. Finally, both similarity measures are combined to an overall similarity measure. The system was used at the Radiology Department at the University of Halle.

Jaulent *et al.* [11] is diagnosing histopathology in the breast cancer domain. Their system uses cases that are derived from written medical reports. A case has an internal tree structure, and represents a collection of macroscopic area. Every macroscopic area is a collection of histological areas, and each histological area contains a cytological description of subjective features, like a big cell size. The features are also weighted for importance. Cases are compared for structural (structure of the histological tree), surface (semantic resemblance of microscopic areas) and feature similarity. A translation transposes the subjective features into numerical values.

CARE-PARTNER [6, 12] is a decision support system for the long term followup of stem cell transplanted patients at Fred Hutchinson Cancer Research Center (FHCRC) in Seattle. The CARE-PARTNER system gives medical and decision support to the home care providers that follow up the transplant patients, using the Internet to connect the home care providers with the FHCRC transplant specialists. The system uses a multi modal reasoning framework, combining Case Based Reasoning and Rule Based Reasoning. A safety insurance plan at three levels (a procedural, a software engineering and a knowledge level) is adopted to ensure fault tolerance. One main characteristic of the system is that it uses a rich knowledge base of prototypical cases and practice guidelines to interpret medical cases and guide the case based reasoning.

Schmidt *et al.* deal specifically with prototypes in [13], where a prototype denotes a generalization occurring as a result of grouping/clustering single cases into more general ones. The claim is made that generating prototypes is also an adequate technique to learn intrinsic case knowledge, especially if the domain theory is weak. Storing new cases may improve the ability to find solutions for similar cases, but to understand the knowledge included within, generalization is needed. Schmidt and Gierl have developed several systems focusing on generalizing into prototypes, as described in their 1998 medical CBR survey [4], such as **ICONS** [14] for antibiotic therapy advice, **GS.52** for diagnosis of dysmorphic syndromes, **COSYL** for liver patient treatment strategies, and **TeCoMED** for forecasting epidemics of infection diseases. These are all further described in [4] and [13]. In [13], Schmidt argues that the reason for using prototypes varies with the type of application and task. In areas where the domain theory is weak, prototypes help to guide the retrieval. In other systems, prototypes may correspond directly with the physicians view and be absolutely necessary for the project. Prototypes also help to speed up retrieval by decreasing the number of cases. The general drawback of prototypes is however loss of information when generalizing.

MED2000. Goodridge *et al.* [7] presents a theoretical diagnostic model for dealing with medical CBR domain problems. The theoretical model, referred to as the Case-Based Neural Network Model, incorporates CBR within a neural network, and the concept of representing knowledge using frames. The CBR-specific problems addressed are all of those mentioned in the introduction, but unfortunately the description lacks a thorough investigation of how, or even if, most of the problems can be remedied with the proposed method. The heart of the model is the separation of case information into two layers, keeping all information identifiers and case features in layer one, and the actual solutions in layer two. Doing this, the system can eliminate the problem of case representation as the information entities in layer one are independent of the solutions. The paper also introduces the concept of pure cases as a way of dealing with the adaptation problem, but it is unclear whether it will actually present an improvement. MED2000 is a hybrid system, has low autonomicity due to experts accepting/declining every hypothesis, and contains a fairly small number of cases, approximately 40 cases.The neural network architecture provides a level of "natural" prototype usage.

7.3.2 Classification systems

Montani *et al.* has focused on CBR in hemodialysis treatments for end stage renal disease [15]. Their system is applied to the efficiency assessments of hemodialysis sessions. Each new dialysis session, i.e. assessment, is represented as a case in the system. Patterns of failures over time, from the patients past history, and cross references with other patients, can be found with this solution. Features are both statically and dynamically collected. The static features are patient information of a general nature (age etc.), and the dynamical features originates from online measurements in the form of continuous time series. The online features used for assessment is mainly derived from the extracorporeal circuit during a dialysis session, like measuring the arterial pressure.

Costello and Wilson [16] is focusing on the classification of mammalian DNA sequences, and are using a case library of nucleotide segments. The stored segments are already classified as exons (carrying information on how to create proteins) and introns (junk segments that do not carry any information). The system is identifying exons in a seemingly random mix of exons and introns in strands of DNA. An edit distance calculation of, insertion, substitution and deletion of individual nucleotides in the tested exons is used to evaluate the similarity between the test strand and the store exon cases. Matched exons is then grouped through activation levels (number of similarities) to find new segments of exons in the test strand.

Nilsson *et al.* [17] address the domain of psychophysiological dysfunctions, a form of stress. The system is classifying physiological measurements from sensors. The system is divided into smaller distinct parts. Measurements, like signals from an ECG, are filtered and improved. A case library of models of distortions etc. is applied to the filters. Features are extracted from the filtered signals (measurements). An additional set of features are extracted from the first set, for trend analysis etc. The features from the first and second set, and patient specific data, are used as a case. The cases are classified with a k-nearest neighbor match.

TeCoMED. Further information about the TeCoMED system was given in [18]. Schmidt and Gierl attempt to use a prognostic model to forecast waves of influenza epidemics, based on earlier observations done in previous years. TeCoMED combines CBR with Temporal Abstraction to handle the problem of the cyclic but irregular behavior of epidemics. Trends are discretized into *enormous decrease*, *sharp decrease*, *decrease*, *steady*, *increase*, *sharp increase*, and *enormous increase*, based on the percentage of change. TeCoMED utilizes former courses and similar cases in a way similar to early kidney problem warnings in the ICONS system. Attempting to commercialize the system, a small software company has incorporated warnings that are generated by the system into web pages of a health insurance scheme and a page of the health authority of the federal state.

Montani *et al.* [19] attempt to integrate different methodologies into a Multi-Modal Reasoning (MMR) system, used in therapy support for diabetic patients. The authors argue that most systems trying to utilize more than one methodology do so only in an exclusive fashion, with methodologies functioning merely as extensions to one another. Montani argues that a MMR system needs much closer integration of technologies to get the full benefits of a multi-modal solution. Integration allows tackling well known problems of single methodologies, i.e. the qualification problem in RBR and the too-small-a-library problem in CBR. The proposed system tries to use a fuller integration and utilize CBR, Rule-Based Reasoning, and Model-Based Reasoning (MBR).

Perner *et al.* [20] has developed a system for the identification of airborne fungi. The fungal strains have a high biological variability, i.e. dissimilarity between the features of individual fungi is quite extensive. A strain can not be generalized to a few cases because of this variability. The images used originate from microscope enhanced pictures. A case is described by attributes (features) derived from the images. Attributes are in the abstraction level of color, shape, size etc. New and original cases (descriptions of individual fungi) are retained in the case library, which is constructed by decision tree and prototype learning methods.

7.3.3 Tutoring systems

WHAT [21] is a tutoring medical CBR system for the education of sports medicine students. WHAT is designed to give better matching exercise prescriptions than the conservative rule-based approach taught by most books. The system provides two separate recommendations for exercise prescriptions, one which is based on the rules found in the books, the other uses CBR with a stored case base made by an expert. The prescribed exercises are applied to cardiac and pulmonary disease patients, as well as issues of general health and lifestyle. The prescriptions are based on features from the patients' medical history and on physiological tests.

Bichindaritz *et al.* [22] have evolved CARE-PARTNER into a medical training system on the Internet. The intention is to help medical students improve their knowledge by solving practice cases. Prototypical cases consist of clinical pathways, which can be tailored to generate cases of varying levels of complexity. The system is also able to evaluate the solutions given by the students for the practice cases. Due to the unlikelihood that a student solution matches the stored solution exactly, a correctness score is calculated and the student solution is placed into one of three categories: Fails to meet standards, Adequate, and Meets all standards.

7.3.4 Planning systems

The **Auguste** project [23], is an effort to provide decision support for planning the ongoing care of Alzheimer's Disease (AD) patients. The first reported system prototype supports the decision to prescribe neuroleptic drugs for behavioral problems. The prototype is a hybrid system where a CBR part decides if a neuroleptic drug is to be given, and a Rule-Based Reasoning (RBR) part decides which neuroleptic to use. The system uses approximately 100 features, manually extracted from medical charts, in each case for determining the right neuroleptic drug. The patient is initially screened for behavioral problems before a Nearest Neighbour match makes a suggestion on whether or not to give neuroleptics to the patient. If the CBR module finds it appropriate to give neuroleptics and no contradictions are found, e.g., allergies to certain drugs etc., the RBR module determines which neuroleptic (of five available) to use. This prescriptive task, although termed "planning" in the vernacular, may be best characterized as one of design.

Davis *et al.* [24] are using a planning system based on the ReCall CBR shell. The system decides what kind of SMARTHOUSE devices disabled and elderly people need in their homes for independent living. Features are constructed from manual translations of written reports. The system contains 10 clustered problem space groups and 14 solution groups. Every group is subdivided by a C4.5 decision tree for efficiency and as an easy way to explain the reasoning process.

7.4 Trends in medical CBR

Naturally, the selection of papers in the previous section is highly subjective. None the less, certain trends are distinctive enough to deserve mentioning.

7.4.1 Property matrix

The research papers used as underlying documentation for the system descriptions does not always contain sufficient information about whether or not a system exhibits a certain construction-oriented property. For completion, the system authors were therefore contacted and asked specifically about the missing property information. Additionally, the authors were asked about the practical use of the systems in everyday life and whether there had been any attempts at commercialization. The answers to the questionnaire are presented in Figure 7.1.

Notably, the majority of systems are multi-modal. Only one of the systems utilizes adaptation. Generalization using prototypes appears to be rare; however, in several projects the intention is to extend the system with prototypes at a later stage. The majority of systems are dependant on some level of user interaction in the reasoning cycle. A few of the systems has been commercialized to some degree, but typically the projects are kept on a research level. Safety and reliability constraints are not too common. Systems that do have safety-critical constraints usually depend on operational reliability as well.

7.4.2 Construction-oriented trends

Looking at the previously defined construction-oriented properties, a number of trends can be identified.

Hybrid systems, also commonly referred to as Multi-Modal Reasoning Systems, constitute the majority of medical CBR systems. The combination of CBR with assisting technologies seems especially successful when CBR acts as the top level coordinator at the system level. Medical systems based on a straight CBR approach may suffer from unreliability, since all reference information is concentrated to the cases. Hence, systems like CARE-PARTNER have built in safeguards.

The autonomicity of the majority of systems is relatively low. Considering the inherent problem of unreliability in CBR, and the fact that systems typically does not reach a 100% correspondence with reported correct solutions even for controlled sets of cases, not relying on complete autonomicity appears to be sound.

Author / System	Cases	Prototypes	Adaptivity	Hybridity
Schmidt/TeCoMED	3000	No (intended)	No (intended)	CBR most imp.
Nilsson/Stress diagnosis	20	No	No	CBR most imp.
Montani/Hemodialysis	1000	No	No	Pure CBR
Montani/Diabetes (MMR)	150	No	No	Hybrid
Costello/Gene finding	948	No	No	Pure CBR
Evans-Romaine/WHAT	25	No (not yet)	No (not yet)	CBR most imp.
Marling/Auguste	28	No	No	Hybrid
Perner/Fungi identification	100	Some extent	No	Pure CBR
Perner/Image segm.	1000	Some extent	No	
El Balaa/FM-Ultranet	130	No		Pure CBR
Bichindaritz/CARE-PARTNER	4000	Some extent	Largely	CBR most imp.

Author / System	Interaction (autonomicity)	Commer-cialisation	Every-day use	Safety criticality	Reliability
Schmidt/TeCoMED	None	Some extent	Largely	No	Soft const.
Nilsson/Stress diagn.	Some extent	No	No	No	No
Montani/Hemodialysis	Some extent	No	No	No	No
Montani/Diabetes	Some extent	No	Some extent	No	No
Costello/Gene f.	None (intended)	No	No	No	No
Evans-Romaine/WHAT	None	No	No	Soft const.	Soft const.
Marling/Auguste	Largely	No	No	Hard const.	No
Perner/Fungi identif.	Some extent	Some extent	Largely		No
Perner/Image segm.	Some extent	Planned	Planned		No
El Balaa/FM-Ultranet	Some extent	No	No	Soft const.	
Bichindaritz/C.-P.	Some extent	No	Some extent	Hard const.	Must work

Figure 7.1: System property matrix. An empty cell denotes that the property could not be determined.

The use of prototypes through case aggregation seems to be a commonly intended future extension, although only partly apparent in the property matrix. Prototypes are already used by many of the systems created by Gierl and Schmidt (as described in Diagnostic Systems), and prototype support is planned for both TeCoMED and WHAT.

7.4.3 Overall trends

The majority of systems in the purpose-oriented category belong to classifying and diagnostic systems. True to the nature of the domain, the emphasis in the medical AI domain has and probably will continue to be on clinical use, i.e., systems involved in some sort of treatment.

Features and feature extraction is an important part of most CBR systems. One identifiable trend in medical CBR is the continuation of separate pre-processing methods on the input data, whether it is a human or an automated process. The datasets are often too large for a direct CBR analysis, and therefore needs to be pre-processed. Examples of systems focusing on separate feature extraction are the stress diagnosis system by Nilsson *et al.* and the airborne fungi detection system by Perner *et al*.

As was the case in the 1998 medical CBR survey by Gierl and Schmidt [4], medical tutoring systems utilizing CBR are rare. The inherent case- and example-based nature and the cognitively plausible model of CBR should be ideal for teaching medical knowledge; still the number of tutoring systems is remarkably low. There is however an increasing number of systems that could partly be seen as tutorial, i.e. the system covers more than one of the purpose-oriented properties, including the Auguste project, WHAT, and FM-Ultranet.

7.5 Conclusions

Although the recent five years has not seen any dramatic changes in the construction and use of medical CBR systems, the field is evolving steadily but slowly. The potential for automated systems in clinics is high, but has yet to reach its full potential. Most systems tend to concentrate on diagnostic tasks, but the use of CBR for therapeutic planning appears to be on the increase. Medical tutoring systems based on CBR are still rare.

The clear majority of systems built around a combination of CBR and other AI methods indicates that most medical domain problems looked into by researchers in recent years have been too complex and multi-faceted to handle using CBR alone. Arguably, hybrid systems have been utilized in the CBR health science domain from the very beginning, with early projects such as CASEY [2] utilizing a mixture of CBR and RBR. There is, however, an increasing majority of hybrid systems being developed, which appears to reflect the increasing complexity and scope of the problem domains.

7.6 Acknowledgments

The authors would like to thank the participants of the questionnaire for the feedback, on which parts of the findings were based.

Bibliography

[1] Edward H. Shortliffe. *Computer-Based Medical Consultations: MYCIN*. Elsevier, North Holland, New York, NY, 1976.

[2] Phyllis A. Koton. *Using Experience in Learning and Problem Solving*. MIT Press, 1988.

[3] Ellis Raymond Bareiss. *Exemplar-Based Knowledge Acquisition: A unified Approach to Concept Representation, Classification and Learning*. UMI, 300 North Zeeb road, Ann Arbor, MI 48106-1346, 1989.

[4] Lothar Gierl and Rainer Schmidt. Cbr in medicine. *Case-Based Reasoning Technology, from Foundations to Applications*, pages 273–297, 1998.

[5] Rainer Schmidt and Lothar Gierl. Case-based reasoning for medical knowledge-based systems. In *German Workshop on Experience Management*, pages 720–725. Proceedings of MIE'00 and GMDS'00, 2000.

[6] Isabelle Bichindaritz. Solving safety implications in a case based decision-support system in medicine. In *Workshop on CBR in the Health Sciences*, pages 9–18. ICCBR'03, June 2003.

[7] Wayne Goodridge, Hadrian Peter, and Akin Abayomi. The case-based neural network model and its use in medical expert systems. In *Artificial Intelligence in Medicine*, pages 232–236. AIMDM'99, June 1999.

[8] Ziad El Balaa, Anne Strauss, Philippe Uziel, Kerstion Maximini, and Ralph Traphner. Fm-ultranet: a decision support system using

case-based reasoning, applied to ultrasonography. In *Workshop on CBR in the Health Sciences*, pages 37–44. ICCBR'03, June 2003.

[9] Ziad El Balaa and Ralph Traphoner. Case-based decision support and experience management for ultrasonography. In *German Workshop on Experience Management*. GWEM'03), April 2003.

[10] Petra Perner. An architeture for a cbr image segmentation system. *Journal on Engineering Application in Artificial Intelligence*, 12(6):749–759, 1999.

[11] Marie-Christine Jaulent, Christel Le Bozec, Eric Zapletal, and Patrice Degoulet. A case-based reasoning method for computer-assisted diagnosis in hisopathology. In *Artificial Intelligence in Medicine*, pages 239–242. AIME'97, March 1997.

[12] Isabelle Bichindaritz, Emin Kansu, and Keith M. Sullivan. Case-based reasoning in care-partner: Gathering evidence for evidence-based medical practice. In *Advances in CBR: 4th European Workshop*, pages 334–345. ECCBR'98, September 1998.

[13] Rainer Schmidt, Bernhard Pollwein, and Lothar Gierl. Experiences with case-based reasoning methods and prototypes for medical knowledge-based systems. In *Artificial Intelligence in Medicine*, pages 124–132. AIMDM'99, June 1999.

[14] Rainer Schmidt, Dagmar Steffen, and Lothat Gierl. Evaluation of a case-based antibiotics therapy adviser. In *Artificial Intelligence in Medicine*, pages 462–466. AIME'01, July 2001.

[15] Stefani Montani, Luigi Portinale, Giorgio Leonardi, and Riccardo Bellazi. Applying case-based retrieval to hemodialysis treatment. In *Workshop on CBR in the Health Sciences*, pages 53–62. ICCBR'03, June 2003.

[16] Edwin Costello and David C. Wilson. A case-based approach to gene finding. In *Workshop on CBR in the Health Sciences*, pages 19–28. ICCBR'03, June 2003.

[17] Markus Nilsson, Peter Funk, and Mikael Sollenborn. Complex measurement classification in medical applications using a case-based approach. In *Workshop on CBR in the Health Sciences*, pages 63–72. ICCBR'03, June 2003.

[18] Rainer Schmidt and Lothar Gierl. Prognostic model for early warning of threatening influenza waves. In *German Workshop on Experience Management*, pages 39–46. GWEM'02, March 2002.

[19] Stefania Montani, Paolo Magni, Abdul V. Roudsari, Ewart R. Carson, and Riccardo Bellazzi. Integrating different methodologies for insulin therapy support in type 1 diabetic patients. In *Artificial Intelligence in Medicine*, pages 121–130. AIME'01, July 2001.

[20] Petra Perner, Thomas Gunther, and Horst Perner. Airborne fungi identification by case-based reasoning. In *Workshop on CBR in the Health Sciences*, pages 73–79. ICCBR'03, June 2003.

[21] Kathleen Evans-Romaine and Cindy Marling. Prescribing exercise regimens for cardiac and pulmonary disease patients with cbr. In *Workshop on CBR in the Health Sciences*, pages 45–52. ICCBR'03, June 2003.

[22] Isabelle Bichindaritz and Keith Sullivan. Generating practice cases for medical training from a knowledge-based decision-support system. In *Workshop Proceedings*, pages 3–14. ECCBR'02, September 2002.

[23] Cindy Marling and Peter Whitehouse. Case-based reasoning in the care of alzheimer's disease patients. In *Case-Based Research and Development*, pages 702–715. ICCBR'01, July/August 2001.

[24] Genevieve Davis, Nirmalie Wiratunga, Bruce Taylor, and Susan Craw. Matching smarthouse technology to needs of the elderly and disabled. In *Workshop on CBR in the Health Sciences*, pages 29–36. ICCBR'03, June 2003.

Chapter 8

Paper C: Building a Case Base for Stress Diagnosis: An Analysis of Classified Respiratory Sinus Arrhythmia Sequences

Mikael Sollenborn, Markus Nilsson

In Proceedings of the Workshop on Health Science at the 7th European Conference on Case-Based Reasoning, Madrid, August 2004

Abstract

Building upon previous work on pattern classification of Respiratory Sinus Arrhythmia (RSA) used in stress disorder treatments, this paper presents an extended approach to analyze the implications of sequences of classified RSA. Through a combination of consulting with field experts and clustering of RSA sequences, each coupled to a particular patient session, session stereotype cases are created to allow for analysis of RSA time series on a structural level, both as a way for physicians to discover previously unknown relationships or verify their own theories and as a way to make the thought process easier to follow for non-experts.

8.1 Introduction

Stress and stress related disorders are today a major concern in the industrialized world, both from a social and an economical standpoint. Yet, the amount of research in the field is far from overwhelming, and much remains to be explored, or even agreed upon. In recent years, psychophysiology - the combined study of the mind and the body and their relationships and ability to affect, consciously or subconsciously, each other, has been strengthened as one of the most promising methods for reasoning about, diagnosing, and treating stress and stress related disorders [1].

Respiratory Sinus Arrhythmia (RSA) is defined as the variation in heart rate (heart beats per minute) that accompanies breathing, known as Heart Rate Variability (HRV). During inhalation, the heart rate increases, and decreases again during exhalation. On a physiological level, the heart rate variation occurs as a result of different activity levels of the sympathetic and the parasympathetic parts of the autonomous nervous system during different stages of the respiration cycle [2, 3].

Physicians studying causes and effects of psychophysiological disorders such as stress and stress related diseases have a valuable tool in the study of RSA and RSA patterns. Its usefulness lies primarily in the ability to help indicate irregular heart rate patterns, some of which may be caused by physical or physiological stress. Most current RSA analysis is done ”on-line”, while the patient is being examined, which is quick but does not allow for a more structural analysis of sequences of RSA patterns. Studying RSA after a session has finished typically involves looking through long time series of continuous data, an often time-consuming and inefficient procedure. Recent research done by Nilsson and Funk [4] has shown the possibility to decrease the burden of the physicians by helping to classify individual respiration periods (a respiration period starts when an exhalation stops and inhalation begins, and ends upon the end of the next exhalation) into one of a set of known RSA dysfunctions, using a case-based approach. This work also showed that the initial set of RSA dysfunctions initially incorporated into the classification system was not complete like it was believed to be, since new dysfunctions were discovered in the process of evaluating the correctness of the classifications.

This paper presents a layered approach to extending the analysis of separated RSA (divided into respiration periods) into studying sequences of classified RSA. The motivation, although taken to a more structural level, is essentially the same as the previous work: to build a knowledge-based system, based partly on expert knowledge and partly on clustering of classified RSA to make assumptions about its meaning, to eventually be implemented as a decision support system.

Case-Based Reasoning (CBR) presents a natural advantage in domains where the underlying domain theory is only partially understood, since the soft matching enables a natural fault tolerance. We use clustering of patient data in the form of classified RSA to establish recurring RSA sequence patterns, then consult domain experts to verify the relationships. Because the aggregated data is not cases but sequences of classified RSA case dysfunctions, and because we make no assumptions about being able to cover the entire domain even by generalization, we prefer referring to the resulting case base as consisting of case stereotypes rather than prototypes.

Combining time-series analysis (in various stages) with Case-Based Reasoning has been studied by Gierl and Schmidt in the ICONS system [5, 6], where a modified Case-Based Reasoning cycle is utilized to forecast kidney functions from measurements and trends from the states. Montani *et al.* [7, 8] uses a combination of CBR, Rule-Based Reasoning (RBR) and Model-Based Reasoning in a Multi-Modal Reasoning system designed to suggest insulin therapy dosages for diabetic patients. For other related material, see the work done by Bichindaritz in the CARE-PARTNER system [9, 10], Marling in the Auguste system [11], and by Perner looking into airborne fungi identification [12].

The rest of the paper is organized as follows. In section 8.2 we take a further look at the motivation for this work and the problems inherent in studying sequences of classified RSA. In section 8.3 we explore the system architecture and the integration of previous work with the presented system approach. Section 8.4 deals with creating clusters and eventually a case base, using expert knowledge and different stages of clustering, and section 8.5 concludes the paper.

8.2 RSA Sequence analysis

In this section we explore different types of RSA, their content, and how sequences of classified RSA can be analyzed.

8.2.1 Normal and dysfunctional RSA

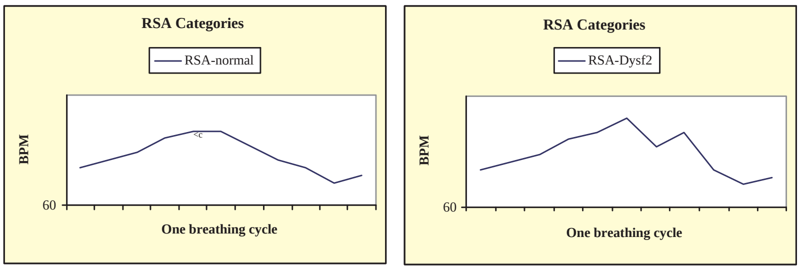

Figure 8.1: RSA category examples: Normal RSA to the left, dysfunctional RSA to the right.

Figure 8.1 shows a normal RSA to the left, and one of many possible dysfunctions to the right. As seen in the picture, the normal RSA resembles a sinus wave, increasing steadily during inhalation and decreasing during exhalation. To the right, an RSA dysfunction that contains a notch towards the end of the curve is depicted. Of course, these depictions are not "exact" - a notch may be of varying size and numbers - but rather illustrates a distinct pattern. In the underlying system, called Hr3Modul, RSA is analyzed on two levels, both looking at the original curve as well as the HRV frequency spectrum obtained through a Fourier transformation. For more details on the RSA classification system, see [4]. For more information about the possible cases of dysfunctions used in the Hr3Modul system, see [13].

8.2.2 Classified RSA sequences

A clinical measurement session done with the Cstress/AirPas systems usually lasts 10-15 minutes, with an average of 5-15 seconds per respiration period. Thus, a single measurement contains on average 60-80 respiration periods (from inhalation to exhalation). Each respiration period is classified into one of the RSA cases from the dysfunctional RSA case base by the Hr3Modul system, leading to a linear sequence of 60-80 RSA categorizations. Looking at these sets, the first thing that needs to be addressed is the relevant frequency of recurring patterns. Although the majority of noise and possible mis-readings in the transformation of measurements to a digital signal in the sensor is assumed to have been taken care of in the earlier stages of the RSA analysis (by Hr3Modul), there is still the possibility that some categorizations may not be dysfunctions but distortions in the sensor readings. Assuming that dysfunctional patterns will tend to repeat themselves, we need to determine the frequency for which to assume that dysfunctions are relevant for the entire session.

The general goal of the clustering process is to identify the possible presence of recurring chains of RSA sequences; that is, to identify dysfunction session clusters that will serve as a basis for experts to further investigate and eventually being able to create generalized RSA session cases to be used in a CBR system. A chain of classified RSA can be seen as a vector consisting of n possible dysfunctions. The difficulty involved in clustering these vectors is primarily the fact that different individuals have different breathing rhythms, which means that the length of the vector (n) will be varying between different individuals. Thus, we cannot perform a direct comparison between vectors but have to rely on comparing parts of the vectors, then modifying the comparison results to reflect differing lengths.

Each measuring session is divided into different phases, concerned with psychophysiological reactions to different types of stimuli, such as baseline measurements (normal breathing), provoking stress, breathing deeply etc. Normally, a session contains 6 phases, always occurring in the same order. Clustering is done both on entire sessions (consisting of phase dysfunctions) and on phases to allow for studying recurring patterns on different levels.

8.3 System architecture

In this section we outline the proposed system and its overall structure. It should be noted that parts of the system have yet to be implemented, and that large-scale evaluations has not yet been done.

8.3.1 Overall system architecture

Figure 8.2 shows a schematic view of the Hr3Modul system integrated with the RSA sequence clustering system. A patient in the system has been subjected to one or more measurement sessions. It is worth noting that each session is treated individually even though a number of sessions may stem from the same patient (that is, no "grouping" of data from the same patient), since in this approach we are not interested in the gradual change of individual patients, but only recurring patterns as a whole. Ideally, we would later want to be able to recognize a patient's gradual change as he gets better or worse by seeing his changes as a number of recurring RSA sequence clusters, i.e., session stereotype cases. A time series session is fed into Hr3Modul, which converts the time series into a sequence of classified RSA and place it in a library of classified RSA sequences. A classified RSA sequence is a vector with additional information about where session phases starts and stops. The clustering module, which is the core of the system, produces a number of session clusters - recurring RSA sequences - using all of the classified RSA sessions produced by Hr3Modul. If the system has been used earlier on and re-clustering is attempted, the clustering module can take advantage (i.e., avoid rejected clusters and re-cluster into accepted clusters) of previous RSA sequences verified by expert physicians. After the clustering is done, expert physicians verify the session clusters created by the clustering module. This process may potentially take a long time and involve a number of new sessions. Physicians may then choose to accept or reject clusters, and accepted clusters are manually merged to form session stereotype cases, which are later to be used in a multi-layered CBR system (see section 8.3.3).

8.3.2 Clustering architecture

The clustering module can be further divided into smaller parts, responsible for clustering phases and for handling the issues outlined in section 8.2.2. Figure 8.3 shows the different parts of the subdivided clustering architecture. All available classified RSA sessions are first sent to the phase clustering module, which divides each session into phases (the beginning and end of each phase is contained within the data). To avoid

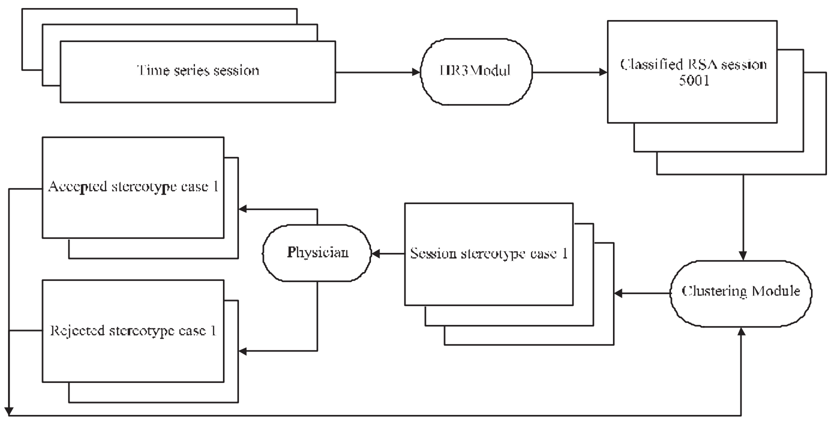

Figure 8.2: RSA sequence identification system overview.

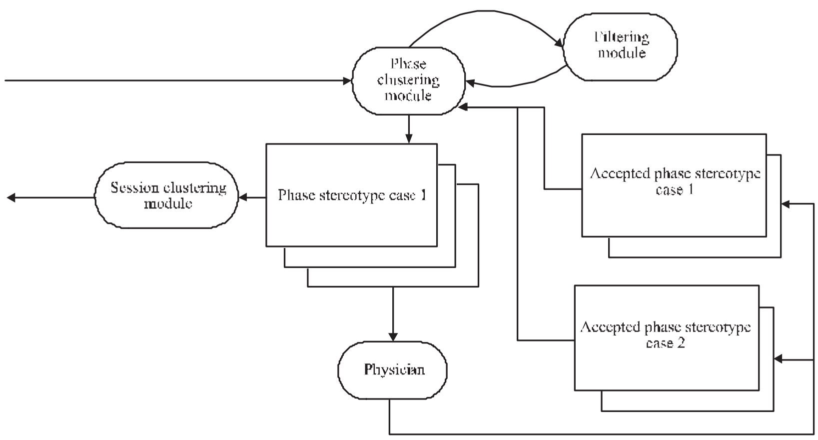

Figure 8.3: Clustering architecture.

ending up with a large number of clusters which are essentially normal but contain a few suspect dysfunctions (which are typically caused by mis-readings in the sensors, as explained in section 8.2.2), phases are sent to the filtering module to remove the dysfunctions from these sessions. The Phase clustering module then clusters each phase with the corresponding phases in all other sessions. The clusters identified are verified by experts (this will certainly be needed on the first clusterings, but on re-clustering it may not be necessary if the phase clusters have been stabilized). Similarly to the general architecture, the experts will divide clusters into accepted and rejected, information which can be used by the phase clustering module when attempting to re-cluster with new sessions. After all phases have been clustered (resulting in a number of indexed phase types - clusters verified by experts), the phases are re-assembled into a new vector containing 6 phase types (each session is typically divided into 6 phases), and the session Clustering Module clusters these vectors as depicted in Figure 8.2.

8.3.3 Utilizing session stereotype cases in a CBR system

After a number of verifications with experts and re-clusterings to make sure that the session clusters reflects valid, recurring, meaningful RSA sequences, the session clusters will be merged by experts into generalized cases. A generalized case, both in the form of phase stereotypes and sessions stereotypes, consists of a context-free grammar defining all variations among the vectors contained in the cluster. The generalized cases are incorporated into a case base as the second layer in a multi-layered stress analysis CBR system with Hr3modul acting on the first level, and the RSA sequence cases being used on the second level. In the end, the two layers are supposed to function as a vital part of a full-scale diagnosis and decision support system for stress and stress disorder analysis. Again, it is important to note that while the individual dysfunctions used as cases by Hr3Modul are well known by the experts, meaningful sequences of RSA are largely unknown and thus can not simply be generated by experts without the aid of a clustering system. Building the case base on layer two is a complex process of clustering and manual verification in several steps, as described in this paper.

8.4 Building the case base

This section deals with clustering of data to produce clusters on different levels of analysis. The clusters are later evolved by physicians into stereotype cases.

8.4.1 Phase clustering

The phase stereotype clustering, where classified RSA sequences are first divided into 6 phases and then clustered individually, is the more difficult of the two clustering stages, since the vectors can be of different lengths. As explained earlier, this is caused by the fact that respiration period times vary among individuals and even between the same individual during different sessions, leading to phases consisting of varying numbers of classified RSA dysfunctions.

The RSA sequences used in the phase stereotype clustering contain information about where each phase starts and stops, making it easy to separate phases from each other.

To perform the actual clustering, we employ a hierarchical, agglomerative approach (as described in e.g. [14]) and the distance function shown in Equation (8.1) below, which has been constructed to allow for comparing classified RSA vectors of non-matching lengths. Note that comparisons between single dysfunctions must be binary, since two dysfunctions are either the same or not, i.e., one type of dysfunction is not "closer" to any other particular type of dysfunction. Eq is a function returning 0 if the elements are identical, and 1 if they are not, and $n = Min(|x|, |y|)$.

$$D = ((\sum_{i=1}^{n} Eq(x_{i+j}, y_{i+k})) + 1)/(n+1) \times Max(|x|, |y|)/n \qquad (8.1)$$

D is then calculated t times, where $t = Max(|x|, y|) - Min(|x|, y|) + 1$. For each iteration through t, j is incremented if $Min(|x|, y|) = |x|$, and k is incremented if $Min(|x|, y|) = |y|$. Finally, the lowest D is chosen to represent the best matching comparison between two phase clusters.

Once the clustering is finished, physicians are consulted to determine the optimal number of clusters and to verify the validity and psychophysiological implications of the data contained within each cluster. This process is not necessarily straightforward and may involve lengthy studies and re-clustering of new sessions by the physicians, but will none the less be a simplified process compared to identifying recurring patterns manually.

Clusters that are determined to contain valid and significant data are put into the list of accepted clusters, and rejected cases are placed separately. Note that in the phase clustering, we are not interested in constructing generalized cases as is the case in the session clustering, but to determine which recurring dysfunctional phases that hold significance and can be re-used to construct simplified session sequences of RSA phases instead of individual dysfunctions. Thus, all cluster information is always preserved to allow for re-clustering.

8.4.2 Session clustering

Once the phase clustering has been finished and the phase clusters have been constructed, each whole session can be reconstructed into a simpler form - a vector containing 6 elements, with each element representing a phase cluster. Thus, the session description is a classified chain of 6 RSA phases.

Once again, we employ a hierarchical, agglomerative clustering method to produce classified session clusters. Because each vector has the same length, the distance function is straightforward. As above, a binary function is used to compare elements of the vector since the number indicating a certain phase stereotype case is simply a case index and not a comparable value. The distance function (2) is shown below, where Eq is a function returning 0 if the elements are identical, and 1 if they are not.

$$d(x,y) = \sum_{i=1}^{6} Eq(x_i, y_i) \tag{8.2}$$

Similarly to the phase clustering described above, the session clustering is followed by a lengthy process of expert physicians converting clusters into generalized session stereotype cases and determining which clusters should be rejected because the cluster does not contain significant data.

8.5 Conclusions

In this paper we have presented an approach to analyze sequences of classified RSA patterns, and in the process creating a case base to be used in a layered CBR system. The project has yet to be fully implemented and evaluated, and building the case base may take some time since its creation is dependant on interaction between creating the cases and allowing experienced physicians to evaluate (both using empirical and analytical methods) the cases. However, a case base consisting of relevant dysfunctional RSA sequences will be a highly useful tool in stress diagnosis and treatment, and the actual creation of the case base will help physicians gain additional knowledge about this largely undocumented area. Thus, the presented approach will help to raise the awareness among physicians about how and why they classify sequences of RSA, which previously has been done mostly on an intuitive basis.

Bibliography

[1] B.H.C. von Scheele and I.A.M. von Scheele. The measurement of respiratory and metabolic parameters before and after incremental exercise on bicycle of patients and control participants: Supporting the effort syndrome hypothesis? *Applied Psychophysiology and Biofeedback*, 24(3):167 –177, 1999.

[2] Tuan Pham Dinh, Helene Perrault, Pascale Calabrese, Andre Eberhard, and Gila Benchetrit. New statical method for detection and quantification of respiratory sinus arrhytmia. *IEEE Transactions on Biomedical Engineering*, 46(9):1161–1165, September 1999.

[3] Marek Malik and John Camm. Heart rate variability -standards of measurement,physiological interpretation,and clinical use. *European Heart Journal*, 17:354–381, March 1996.

[4] Markus Nilsson and Peter Funk. A case-based classification of respiratory sinus arrhythmia. In *Proceedings of the The Seventh European Conference on Case-Based Reasoning*. ECCBR'04, 2004.

[5] Rainer Schmidt and Lothar Gierl. Temporal abstractions and case-based reasoning for medical course data: Two prognostic applications. *Machine Learning and Data Mining in Pattern Recognition volume 2123 of Lecture Notes in Computer Science*, pages 23–34, 2001.

[6] Rainer Schmidt, Dagmar Steffen, and Lothat Gierl. Evaluation of a case-based antibiotics therapy adviser. In *Artificial Intelligence in Medicine*, pages 462–466. AIME'01, July 2001.

[7] Stefania Montani, Paolo Magni, Riccardo Bellazzi, Cristiana Larizza, Abdul V.Roudsari, , and Ewart R.Carson. Integrating model-based decision support in a multi-modal reasoning system for managing type 1 diabetic patients. *Artificial Intelligence in Medicine*, 29:131–151, 2003.

[8] Stefania Montani, Paolo Magni, Abdul V. Roudsari, Ewart R. Carson, and Riccardo Bellazzi. Integrating different methodologies for insulin therapy support in type 1 diabetic patients. In *Artificial Intelligence in Medicine*, pages 121–130. AIME'01, July 2001.

[9] Isabelle Bichindaritz, Emin Kansu, and Keith M. Sullivan. Case-based reasoning in care-partner: Gathering evidence for evidence-based medical practice. In *Advances in CBR: 4th European Workshop*, pages 334–345. ECCBR'98, September 1998.

[10] Isabelle Bichindaritz, Carol Moinpour, Emin Kansu, Gary Donaldson, Nigel Bush, , and Keith M.Sullivan. Case based reasoning for medical decision-support in a safety critical environment. *Artificial Intelligence in Medicine, AIME'03*, pages 314–323, November 2003.

[11] Cindy Marling and Peter Whitehouse. Case-based reasoning in the care of alzheimer's disease patients. In *Case-Based Research and Development*, pages 702–715. ICCBR'01, July/August 2001.

[12] P. Perner, T. Gunther, H. Perner, G. Fiss, and R. Ernst. Health monitoring by an image interpretation system - a system for airborne fungi identification. *In Medical Data Analysis volume 2868 of Lecture Notes in Computer Science*, pages 62–74, 2003.

[13] Bo von Schele. Classification systems for rsa, etco2 and other physiological parameters. PBM Stressmedicine, Heden 110,821 31 Bollns, Sweden, 1999.

[14] A.K. Jain and R.C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.