SIMILARITY DETERMINATION AND CASE RETRIEVAL IN AN INTELLIGENT DECISION SUPPORT SYSTEM FOR DIABETES MANAGEMENT

A thesis presented to

the faculty of

the Russ College of Engineering and Technology of Ohio University

In partial fulfillment

of the requirements for the degree

Master of Science

Donald Walker

November 2007

This thesis entitled

SIMILARITY DETERMINATION AND CASE RETRIEVAL IN AN INTELLIGENT DECISION SUPPORT SYSTEM FOR DIABETES MANAGEMENT

by

DONALD WALKER

has been approved for

the School of Electrical Engineering and Computer Science

and the Russ College of Engineering and Technology of Ohio University by

Cynthia R. Marling

Associate Professor of Electrical Engineering and Computer Science

Dennis Irwin

Dean, Russ College of Engineering and Technology

WALKER, DONALD, M.S., November 2007, Computer Science

Similarity Determination and Case Retrieval in an Intelligent Decision Support System for Diabetes Management(109 pp.)

Director of Thesis: Cynthia R. Marling

This thesis presents a metric for similarity determination and case retrieval for an intelligent decision support system. This system may greatly reduce the burden of diabetes management for both diabetic patients and their physicians through the use of case-based reasoning. Diabetes is a disease which affects over 20 million Americans with almost as many being at high risk of developing the disease. In order to live a healthy life with diabetes, individuals must continuously regulate their blood glucose levels. The current state of the art of diabetes management requires frequent doctor visits, careful measurements of the blood glucose levels, and mathematical calculations of insulin doses by the patient. This research is an intermediate step toward development of a method and system to analyze a patient's current state, recognize problems in blood glucose control and suggest therapy adjustments to remedy those problems.

Approved: ____

Cynthia R. Marling

Associate Professor of Electrical Engineering and Computer Science

Acknowledgments

I would like to express my sincere gratitude to all of my friends and family members for all of their support during my time working on this research and beyond. Particularly, I thank my parents, Debra and Mike, and my wonderful girlfriend, Blair, for their unending support in every way throughout all of my endeavors. I would also like to thank all of my professors, especially my thesis committee members, and everyone else involved in my education without whom I never would have made it this far. A special thank you goes out to Dr. Cynthia Marling, Dr. Frank Schwartz and my fellow researchers involved with this project for their generous investment of time and effort in keeping this project going and allowing me to be involved with it.

Table of Contents

List of Figures

Chapter 1

INTRODUCTION

This thesis presents a metric for similarity determination and case retrieval for an intelligent decision support system for diabetes management. This system is intended to greatly reduce the burden of diabetes management for both diabetic patients and their physicians through the use of case based reasoning (CBR). This research is an intermediate step toward development of a method and system to analyze a patient's current state, recognize problems in blood glucose control and suggest therapy adjustments to remedy those problems.

In a CBR system, knowledge is represented in cases, which consist of a problem description, solution to the problem and the outcome of applying the solution to the problem(Kolodner, 1993). In order to solve new problems, the most applicable case must be retrieved from the case library, which represents the system's memory of previously encountered problems. The primary contribution of this thesis is a means for finding the most similar problem, and therefore the most applicable solution, to solve the newly encountered problem with blood glucose control. An additional contribution, made in collaboration with fellow graduate student Eric Flowers, is the design and implementation of a case structure for representing knowledge.

Diabetes is a disease which causes the body to either not produce insulin or to not properly make use of the insulin that it does make (American Diabetes Association, 2002). When carbohydrates are consumed, the body must convert glucose into energy that can be used to do everyday tasks. Insulin is a hormone which facilitates this process and is therefore an important part of the body's normal functioning that helps to maintain an appropriate level of glucose in the blood. In the case of diabetic patients who do not produce or properly use insulin, blood glucose levels must be manually regulated or the patient may suffer undesired consequences (Stratton et al., 2000). Too much glucose in the blood for long periods can cause long term problems, including the risk of stroke and heart disease, while too little glucose in the blood may cause acute conditions, including diabetic coma.

The current state of diabetes management is quite an involved process. A diabetic patient must constantly monitor his or her blood glucose levels in one or more ways. Patients draw blood, usually from the fingertips, several times a day to be tested in a glucose meter. Patients may also wear a continuous glucose monitoring system which provides measurements every five minutes (Medtronic MiniMed, 2007c). When the patient detects that his or her blood glucose level is out of the range specified by their physician, corrective action must be taken to bring the glucose level back into the desired range. Generally, if the blood glucose level is too low, the patient will consume some amount of carbohydrates based on the current glucose level and the patient's individual physiology. If the blood glucose is too high, the patient will usually administer some amount of insulin, which is also determined by current glucose levels and patient physiology.

Diabetic patients need to track their blood glucose levels and note instances of very high and very low levels (hyperglycemic episodes and hypoglycemic episodes, respectively), either electronically or in a log book. Diabetic patients should also visit their physicians three or four times per year to review the trends in their glucose levels and make any necessary adjustments. During these visits the physician must review the patient's blood glucose data, including trends in glucose levels, hyper- and hypoglycemic episodes, and general management habits from the previous months between visits. Due to this large amount of data that must be reviewed during the time of a normal office visit, the physicians experience data overload and run the risk of possibly missing problems in the patient's glucose management.

This research seeks to alleviate that problem through the use of artificial intelligence, specifically case based reasoning (CBR). CBR is an approach that tries to solve newly encountered problems by applying the solutions learned from solving problems encountered in the past (Kolodner, 1993). This is similar to the way a human might solve a newly encountered problem; the human recalls a problem he or she encountered in the past and, depending on whether the solution attempted for that problem was successful or unsuccessful, either applies or avoids that solution for the current situation.

This interdisciplinary study is important to both those in the artificial intelligence (AI) community as well as those in the medical community and is a collaborative effort between Ohio University's School of Electrical Engineering and Computer Science in the Russ College of Engineering and Technology and the Appalachian Rural Health Institute Diabetes/Endocrine Center of the College of Osteopathic Medicine. An Ohio University Institutional Review Board approved study has been completed involving twenty patients with Type 1 diabetes. Data was collected from the patients regarding their lifestyle and glucose management during each patient's six week involvement (Maimone, 2006; Marling et al., 2007a; Marling et al., 2007b; Marling et al., 2007c).

This data was stored in a patient database, reviewed by two physicians specializing in diabetes care, and used to create a case library. This library consists of cases which represent glucose management problems, the physician recommended solutions, and the outcomes of applying said solutions. This case library serves as the memory of "past experiences" for the intelligent decision support for diabetes management (IDSDM). When a new problem is encountered and presented to IDSDM, the similarity determination and case retrieval module searches the case base and locates the case with the problem most similar to the newly input problem. If the cases are similar enough, IDSDM will recommend the solution that was used for the returned case for solving the problem of the new case.

Studies indicate that storing medical information electronically in the form of electronic medial or health records (EMRs and EHRs) may be becoming more popular and nearing widespread usage (Berner et al., 2005; Ash and Bates, 2005; Middleton et al., 2005). This may be due to the fact that the current generation has grown up in an age when computers are viewed as a common household appliance. This familiarity with computers and electronic health resources is spawning a growing trend of users turning to sources such as the Internet to find answers to health questions (Ybarra and Suman, 2006). These studies, along with positive feedback received from the patients (Maimone, 2006), suggest that diabetic patients would find a system such as the finished IDSDM to be a valuable, viable tool for helping with blood glucose management.

Chapter 2 of this thesis presents background information about diabetes, CBR and the IDSDM study. Chapter 3 provides details about the similarity determination and case retrieval metric used in IDSDM. Chapter 4 describes the methodology and results of the two-phase evaluation process for the similarity determination and case retrieval metric and provides avenues for future work for the IDSDM system. Chapter 5 presents examples of related research in the fields of CBR, both medical and non-medical, and diabetes care. Chapter 6 gives the summary and conclusions drawn from the work.

Chapter 2

BACKGROUND

This section will provide some background information regarding topics pertinent to this research. Topics include:

- Diabetes, its effects, who it affects and how it is managed
- The current state of diabetes management
- How intelligent decision support can help
- Case-based reasoning as an approach to intelligent decision support
- The focus of this research

Information regarding related research, including the use of artificial intelligence in the medical field, both medical and non-medical case-based reasoning systems as well as the current state of the art for diabetes management technology, may be found in Chapter 5.

2.1 Diabetes

2.1.1 What is Diabetes?

According to the American Diabetes Association, "diabetes is a disease in which the body does not produce or properly use insulin. Insulin is a hormone that is needed to convert sugar, starches and other food into energy needed for daily life. The cause of diabetes continues to be a mystery, although both genetics and environmental factors such as obesity and lack of exercise appear to play roles" (American Diabetes Association, 2002). These sugars and starches, glucose, accumulate in the blood if not used by the body for energy. Glucose levels are affected by many factors including diet, exercise, insulin, stress and illness. In individuals without diabetes, glucose levels are regulated naturally. Individuals with diabetes, however, face difficulties in glucose management. There are three main types of diabetes mellitus: Type 1 diabetes, Type 2 diabetes and gestational diabetes.

People with Type 1 diabetes, also called juvenile-onset diabetes mellitus or immune-mediated diabetes mellitus, do not produce enough insulin to adequately maintain a normal blood glucose level. This is often due to their immune system mistakenly destroying pancreatic beta cells that produce insulin. Other possible factors for Type 1 diabetes include genetics (family history), viruses, cow's milk at very young ages, oxygen free radicals and chemicals and drugs.

In Type 2 diabetes, also called adult-onset diabetes mellitus or insulin resistant diabetes mellitus, the body either does not produce enough insulin to normally regulate blood glucose levels or insulin receptors are resistant to the insulin that is present or a combination of both. Type 2 diabetes tends to affect the overweight, particularly those who carry their extra fat above their hips. Possible factors for Type 2 diabetes include genetics, age, obesity, and lifestyle. This type of diabetes accounts for between 90-95% of all diabetes cases, one third of whom are unaware of their condition.

Gestational diabetes occurs during pregnancies among women who show no previous signs of Type 1 or Type 2 diabetes. Bodies of people with gestational diabetes produce insulin, but their cells are resistant to it. All pregnant women are at least somewhat insulin resistant due to the hormones produced by the baby. Other possible contributing factors to gestational diabetes include genetics and obesity. Up to 40% of women who get gestational diabetes during pregnancy will be diagnosed with Type 2 diabetes after their pregnancy.

2.1.2 Who Has Diabetes?

Diabetes is a disease that affects many people. Around 20 million Americans (that's more than one in 16) have some type of diabetes mellitus, with 2,200 new cases being diagnosed each day and one million new cases each year. In addition to those diagnosed with diabetes, another 16 million Americans have a condition called pre-diabetes. Individuals with pre-diabetes have higher-than-normal blood glucose levels, but not high enough to be considered diabetic. Nearly 16 million are diagnosed with Type 2 diabetes, with another 6 million suspected to be undiagnosed. Twenty percent of persons over age 65 also have Type 2 diabetes. 850,000 to 1.7 million people have Type 1 diabetes. 135,000 women will be diagnosed with gestational diabetes during pregnancy; 40% of whom will then be diagnosed with Type 2 diabetes (American Diabetes Association, 2002).

2.1.3 What Are The Effects of Diabetes?

Complications of diabetes include both short-term, acute problems, as well as long-term, chronic problems. Among the former group are problems such as diabetic ketoacidosis, nonketotic hyperosmolar coma, hypoglycemia, and diabetic coma. Among the latter group, usually associated with chronically high glucose levels, are diabetic retinopathy, diabetic neuropathy, diabetic nephropathy, coronary artery disease, stroke, peripheral vascular disease, diabetic myonecrosis, and carotid artery stenosis.

Diabetic ketoacidosis is caused by the accumulation of ketones, the by-product of the breakdown of fat cells. This occurs when too little or no glucose is available as an energy source and fat is used instead. During ketoacidosis, the blood becomes more acidic than the body tissue as a result of the extra ketones. This acidity may cause cell damage which could lead to serious illness or, in some cases, death.

Nonketotic hyperosmolar coma occurs during extreme hyperglycemia when water is scarce in the body. Whereas the excess glucose would normally leave the body via urination, the kidneys try to conserve water causing the glucose to remain in the body. This leads to a cycle of dehydration leading to increased blood glucose levels which leads back to dehydration and so on. This condition may lead to shock, cerebral edema, blood clots, lactic acidosis and coma.

Diabetic nephropathy is a condition in which the kidneys cease to function properly, resulting in increased protein levels in the urine. This condition may lead to high blood pressure, chronic kidney failure and end-stage kidney disease.

Diabetic neuropathy is a diabetes complication in which nerve damage results from decreased blood flow and chronic hyperglycemia. This condition affects approximately 50% of diabetic patients. Diabetic neuropathy may lead to constant, intense pain or total loss of sensation in the affected area.

Diabetic retinopathy is a complication that affects the eye's retina. This is caused by damage to the blood vessels in the eye and may lead to blindness and is the leading cause of blindness in working age Americans. Nearly everyone who has diabetes for more than 30 years will exhibit symptoms of diabetic retinopathy.

Most of the complications of diabetes mellitus may be avoided by maintaining normal blood glucose levels (Stratton et al., 2000; The Diabetes Control and Complications Trial Research Group, 1993). Therefore maintaining normal blood glucose levels is the focus of diabetes management.

2.1.4 Diabetes Management

There is no cure for diabetes. As such, the main task of managing diabetes is to keep the blood glucose level within a specified range to avoid short-term emergency problems stemming from hypoglycemia (blood sugar too low) and possible long-term complications of hyperglycemia (blood sugar too high). Blood glucose levels represent the amount of sugar, or glucose, in a person's blood stream at a given time. The American Diabetes Association recommends guideline levels of blood glucose during certain times of the day. The ADA recommends a blood glucose level of 90 milligrams per deciliter (mg/dl) during fasting, 105mg/dl prior to meals, 130mg/dl one hour after meals, and 120mg/dl two hours after meals (American Diabetes Association, 2002).

Keeping blood glucose levels under control can be an arduous task for the diabetic patient, a task which involves, among other things, carefully managing diet, exercising, taking oral diabetes medication, and using some form of insulin. Other variables may also cause fluctuations in blood glucose levels, such as stress, illness, menses, injection site scarring, and other physiological factors unique to each patient. All of these factors combine to make glucose level management a difficult task; one which presents various difficulties for each case. As such, the above ADA glucose level guidelines are just that, merely guidelines. Each patient reacts differently to treatments, thus maintaining frequent interaction with a physician is important to the management process.

Maintaining frequent interaction with a physician, in the case of most diabetic patients, involves making an office visit once every three or four months. During these visits, the patient and the physician review the patient-kept records of blood glucose values, insulin dosages, and any other factors pertinent to diabetes management. If a patient takes a minimum of three fingerstick glucose readings per day, for three months, the doctor has 270 data points to review excluding any extra fingerstick measurements, CGMS data, or other related factors (see Figure 2.1). With that much information to review in the time of a normal office visit, physicians are experiencing data over load: being overwhelmed by the amount of information. In a situation like that, many problems could potentially be overlooked. This study seeks to aid in the analysis of that data. Currently, the focus of the study is on Type 1 diabetes decision support; however, the information gained by this project may be applied to the management of blood glucose for patients with any of the aforementioned types of diabetes.

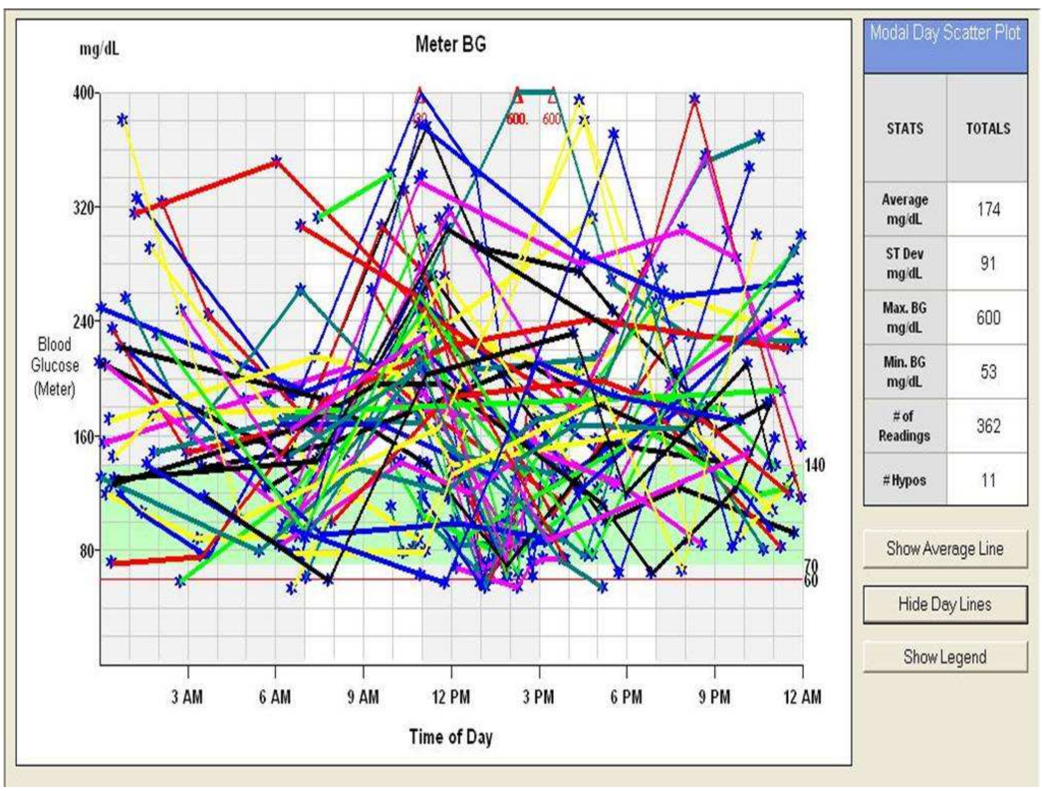

Figure 2.1: 12 weeks of patient glucose data.

2.2 Artificial Intelligence

John McCarthy, who first coined the term "artificial intelligence" (AI) defined intelligence as "the computational part of the ability to achieve goals in the world" and artificial intelligence as "the science and engineering of making intelligent machines, especially intelligent computer programs" (McCarthy, 2000). Research in this field generally seeks to automate tasks which require intelligent thought or behavior. Rich and Knight posit that "Artificial Intelligence is the study of how to make computers do things at which, at the moment people are better." (Rich and Knight, 1991).

Research within the AI field encompasses many sub-fields. These narrower categories include topics such as neural networks, fuzzy logic, genetic algorithms, computer vision, machine learning, expert systems, Bayesian networks as well as case-based reasoning (CBR). The latter is the focus of this research.

2.2.1 CBR

Case-based reasoning, or CBR, is a technique for solving new problems by applying the solution of past problems. Past situations are stored in cases, which represent knowledge related to the various aspects of the situation. A case consists of three major parts: the problem/solution description, the solution and the outcome (Kolodner, 1993). Kolodner defines the first as "the state of the world at the time the case was happening and, if appropriate, what problem needed solving at that time," the second as "the stated or derived solution to the problem specified in the problem description, or the reaction to its situation" and the last as "the resulting state of the world when the solution was carried out." An example of real-world case-based reasoning would be a mechanic repairing a vehicle by recalling a car which displayed similar symptoms and applying the same solution to the new vehicle.

When a new problem is encountered that must be solved, a CBR cycle is executed. Aamodt and Plaza (Aamodt and Plaza, 1994) state that "at the highest level of generality, a general CBR cycle may be described by the following four processes:

- 1. RETRIEVE the most similar case or cases
- 2. REUSE the information and knowledge in that case to solve the problem
- 3. REVISE the proposed solution
- 4. RETAIN the parts of this experience likely to be useful for future problem solving"

In the above example, a mechanic may encounter a car with deceleration problems and recall a past problem, or case, where a vehicle had similar issues. This recollection represents the first step in the CBR cycle. The problem portion of the case will consist of all the details representing the state of the vehicle experiencing poor deceleration. The solution for the past problem represents the details of the fix that the mechanic did to the old vehicle, for instance, changing worn brake pads. The outcome represents the success or failure of the brake change. For instance, a customer brings in a vehicle which has stopping problems and he remembers that the last time he saw a vehicle with stopping problems changing the brake pads fixed the problem. The mechanic then executes step two of the CBR cycle, reusing the information gained from the previous case. At this point, the mechanic may apply the optional step 3 in the CBR cycle: revising the solution. Perhaps the new vehicle has drum brakes rather than disk brakes and needs new brake shoes rather than brake pads. The mechanic then executes step 4, storing the new information gained from this case, namely, the difference between disk and drum brakes.

Humans use this type of problem solving process all the time without being conscious of it (Kolodner, 1993). This approach to problem solving comes naturally to us; we do not think of the process as remembering the past "cases" and applying the learned solutions to a new problem. We may not even be aware that we are implementing a process at all but merely using experience. Using this line of thought fits nicely with Rich and Knight's definition of AI.

2.3 Intelligent Decision Support via Case-Based Reasoning

2.3.1 How Intelligent Decision Support Can Help

As mentioned earlier, physicians are being overwhelmed by the amount of data that they must sift through in the short time they have for an office visit. With the number of patients that each physician tends to on a daily basis it would be nearly impossible to go through each tidbit of data looking for trends or problems. This can lead to potential problems being overlooked and, given the aforementioned complications of diabetes, these oversights may be costly.

The benefits of the intelligent decision support for diabetes management system (IDSDM) are twofold. Firstly, the system collects and organizes data and is capable of displaying it in an orderly fashion. This allows the doctor to visualize the patient's data much more easily than by looking at the patient's handwritten notes for finger-stick readings and general health information (see Figure 2.2). By being able to easily visualize several weeks of patient data in such an organized manner, the physician is more likely to spot emerging trends in the patient's glucose management and prevent possible problems before they get worse.

The second way in which our system helps is the crux of this research. The system will be able to provide intelligent decision support by analyzing incoming data and comparing it to past data that it has already seen. By doing this, the system can match incoming patterns to known past problems, potentially faster than a physician alone could.

2.3.2 The Preliminary Clinical Study

Researchers involved with IDSDM have undertaken a study approved by the Ohio University Institutional Review Board (IRB) which involves 20 Type 1 diabetic patients. During each patient's six week involvement they are asked to provide exten-

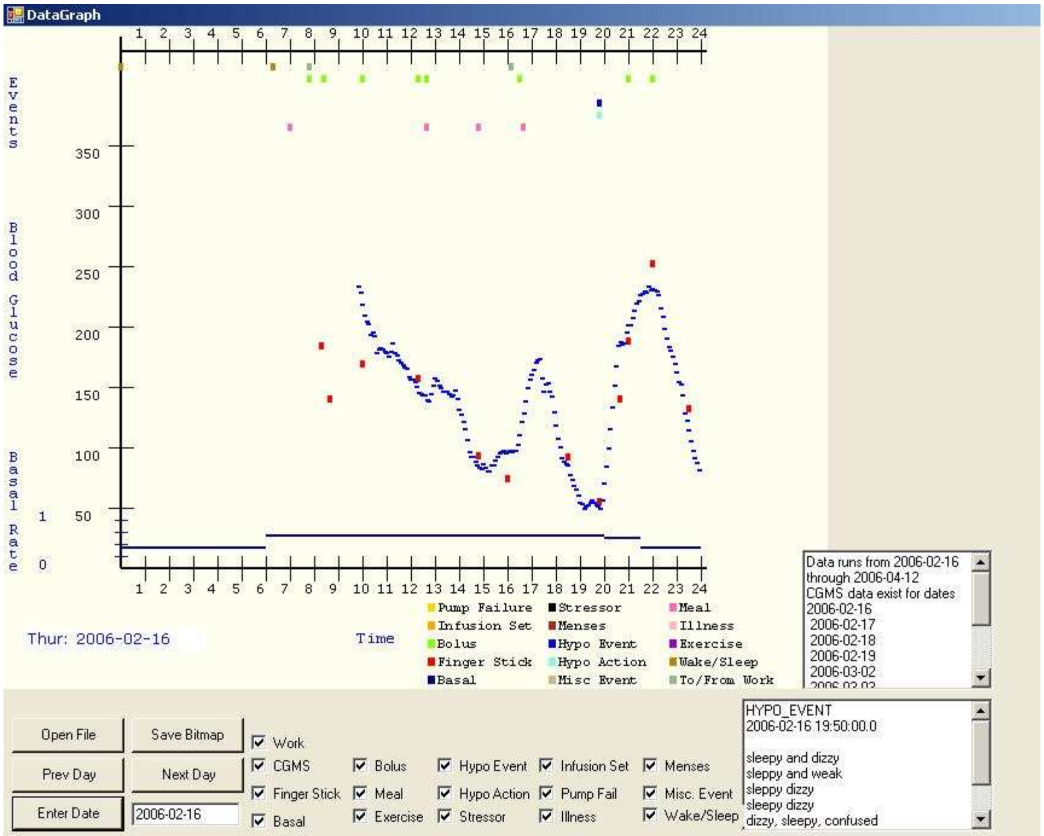

Figure 2.2: DataGraph visualization program. (Wesley Miller)

sive daily logs via the Internet. These logs track glucose levels, insulin dosages, work schedules, sleep patterns, exercise, meals, stress, infusion set changes and other data thought to be relevant to diabetes management.(Marling et al., 2007c) This project is a collaborative effort between the School of Electrical Engineering and Computer Science in the Russ College of Engineering and Technology and the Appalachian Rural Health Institute Diabetes/Endocrine Center of the College of Osteopathic Medicine. To date, the project has included the efforts of five EECS graduate students under the supervision of faculty advisors Drs. Cynthia Marling, Ph.D., and Frank Schwartz, M.D. The goal of the study is to provide intelligent automated decision support to diabetic patients and their physicians to aid in the management of their diabetes.

As mentioned above, there are two main stages of this project. The first part is the data collection (see Figure 2.3), organization and display aspect. This portion of the software was created by Anthony Maimone and Wesley Miller (Marling et al., 2007a; Marling et al., 2007c; Maimone, 2006).

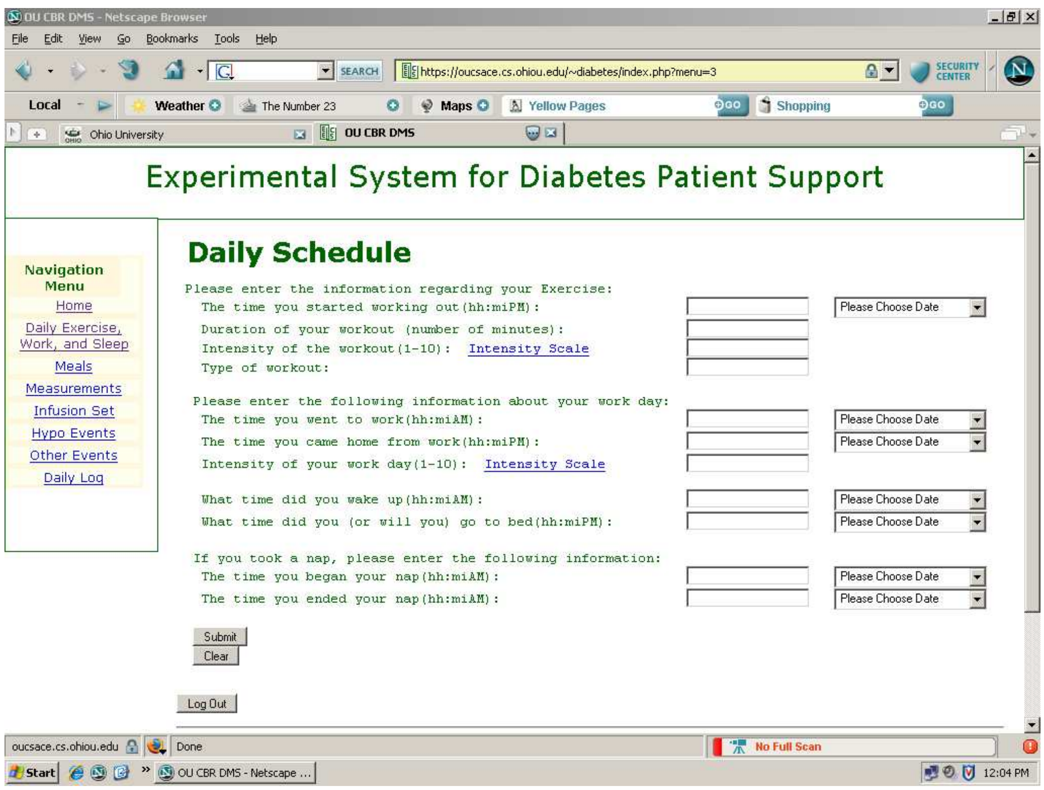

Figure 2.3: Data entry screen on project website. (Anthony Maimone)

The second part is the intelligent decision support aspect. This portion is a joint effort of fellow graduate students Eric Flowers, Thomas Jones, and myself. The way that we provide this decision support revolves around the notion of case-based reasoning (CBR). As described above, the idea behind CBR is for the reasoner to remember past situations and apply the lessons learned therein to the current situation. For purposes of this study, the past situations are problems with blood glucose control and the current situation is the new data being received from the patient, which is what is collected and stored in the first stage of the project. Once a new potential problem is detected, the case-base (collection of past memories) is searched for situations closely resembling the new problem. These past cases are retrieved based on the degree of matching for many factors as well as the importance of that factor to the match. Once similar cases are retrieved, they are tailored to the current patient's physiology by making minor adjustments to the previous problem's solution if needed and then that solution may be applied to the new situation.

This system provides tremendous benefit over the traditional model of the physician only going over the patient's data once every three months. One great help is that the system can run continuously to monitor the patient's current state, picking up any potential problems as soon as they manifest. This early detection allows the doctor to advise the patient on how to remedy the problem and avoid any complications that may be associated with that problem. Another great benefit is the vast amount of memory that the system can have available for storing past cases. While there is a point of diminishing returns for the number of cases in the case base, the computer can positively store and identify problems that may slip the physician's mind, say if he or she had not seen an instance of a particular problem in some time.

2.3.3 The Focus of This Research

The focus of this research deals primarily with the similarity determination and case retrieval aspect of the CBR process (step one of Aamodt and Plaza's generalization). As mentioned in the previous section, a case in a CBR system consists of three parts: problem, solution and outcome. In this system, the problem portion represents the details of specific problems in blood glucose control. The solution represents physician recommended solutions to the management problems. The outcome portion represents the factors relating to the success of the proposed solution and the patient's adherence to that advice. All cases in the case base will have all three sections completed. New cases, however, are not yet solved and will consist solely of the problem portion.

Although several methods of similarity determination were explored, the K-Nearest Neighbor algorithm is implemented for this project. This algorithm involves computing an *aggregate match score*, which is obtained for any two cases by finding the corresponding features of the cases, comparing the values to compute the degree of match and then multiplying by a coefficient representing the importance of the feature to matching.(Kolodner, 1993) This method is used to compute the scores for the similarity of each case in the case base to the new problem. Once the score is computed for the similarity of each case in the case base to the new case, one or more of the top matching cases may be chosen. More details on case structure and similarity determination algorithms are given in the next chapter.

Chapter 3

SIMILARITY DETERMINATION AND CASE RETRIEVAL

The focus of this research deals with the similarity determination and case retrieval aspect of the CBR process (step 1 in the CBR process as described by Aamodt and Plaza in section 2.2.1). This chapter will provide an overview of similarity assessment and the retrieval of cases matching new input as well as describe in detail the process of these steps in the IDSDM system.

3.1 Case Retrieval, Matching and Ranking

3.1.1 Case Retrieval Overview

Retrieval plays a pivotal role in the CBR cycle and, as such, much research has been focused on retrieval and similarity assessment (Lopez de Mantaras et al., 2006). Kolodner specifies that case retrieval generally consists of two substeps: recollection of potentially relevant previous cases and selection of the best subset of those cases (Kolodner, 1993). The aim of the first substep is to "narrow down" the case base to potentially similar cases that may provide relevant solutions to the new case. This may be done by selecting cases which are similar based on only the most important features of the case. This subset represents the cases which most closely match the new case and deserve more "intensive consideration." The aim of the second substep is to choose a single case or small subset of cases from those potentially similar cases found in substep one. These cases are used to determine a solution for the original problem.

Both substeps are not required for the retrieval process. As Jain notes, the retrieval process may either be a one- or two-step process (Jain, 2002). In the one-step process, the first substep is omitted and each case in the case library is compared to the new case and ranked. This means that the subset normally obtained from Kolodner's first substep is simply the entire case-base. Jain describes the situations in which the one-step process is useful as those in which "the case base is small and the domain is weak [not easily understood or ... easily abstracted]." The two-step process uses both of Kolodner's substeps.

Kolodner states that there are several big problems which stand in the way of retrieval, each of which must be addressed before the problem can be solved (Kolodner, 1993). These several problems are collectively called "the indexing problem" and consist of the "similarity-assessment" problem, the "indexing-vocabulary" problem, the "situation-assessment" problem and the problem of creating "retrieval algorithms".

The similarity-assessment problem is the problem of getting the computer to recognize the fact that a case is applicable to a new situation. To go back to a definition of AI from section 2.2, "Artificial Intelligence is the study of how to make computers do things at which, at the moment people are better."(Rich and Knight, 1991) Mantaras et al. state that CBR is fundamentally related to research in analogical reasoning (basic human mechanisms such as matching and retrieval) and go on to say that "there is ample evidence that, for people, retrieval is heavily influenced by surface properties more than by deep similarities, unlike most CBR systems."(Lopez de

Mantaras et al., 2006). Kolodner posits that occasionally the most applicable cases do not appear similar based on surface features (Kolodner, 1993). The example that Kolodner gives is that, based on surface features, chess and football appear dissimilar (where they are played, type of players, number of players, movement, etc.). However, when more abstract features of the two are investigated, they may appear quite similar (both are competitive games involving planning and counter-planning in which each side wants to win and to see their opponent lose, etc.).

Overcoming this problem to get the computer to perform as a human would gives rise to the indexing-vocabulary problem: the determination of abstract terms in which we can describe a case to accurately represent a case such that it may be compared to another case. Much of the early research in the CBR field was focused on the development of these vocabularies (Schank and Osgood, 1990; Domeshek, 1992; Leake, 1992). By ensuring that the case description contains most of the relevant vocabulary, the system can save computation time in using surface features to derive deeper attributes or vice versa.

The next problem facing retrieval is the situation-assessment problem. At the root of this problem lies the need for elaboration capabilities for the computer (Kolodner, 1993). To solve the situation-assessment problem, the system must be able to compile the relevant information about a situation into a case that may be used for comparison. Often this process involves using surface features of a situation to come up with derived features, which may be computationally expensive to determine but necessary for comparison.

The final road block comprising the indexing problem is the issue of creating appropriate retrieval algorithms. The problem here is the need to search a case library of indeterminate size in an efficient manner and return relevant results. This retrieval is accomplished by matching and ranking procedures, which is the crux of this research and is explained in more detail in the following section.

3.1.2 Matching and Ranking Overview

> "Matching is the process of comparing two cases to each other and determining their degree of match. Ranking is the process of ordering partially-matching cases according to goodness of match or usefulness." (Kolodner, 1993)

In a CBR system the returned cases must not only have similar features, but they must be "usefully similar" for solving the problem presented by a new case. Researchers in the pattern matching community define mapping and matching algorithms to determine the degree of similarity between two cases, but one must also consider how important it is that each factor of a case matches in judging similarity (Kolodner, 1993). There are a number of ways to achieve this desired result including various memory organizations and ways of searching those memory structures; several of these methods will be discussed in this section.

Flat Memory with Serial Search

One type of logical storage structure in which CBR systems may keep their cases is known as a flat memory structure. CBR systems which make use of flat memory structures keep their cases stored in a sequential manner in an array or list type structure. Generally, there is no real organization to this type of list; rather, the matching heuristics do all of the retrieval work.

Flat memory structures naturally lend themselves to serial search: application of the matching function to each case in the list or array. As the function compares the new case to each of the existing cases, the results of each comparison are stored, and the case which most closely resembles the newly input case is returned.

Kolodner cites several advantages and disadvantages to the flat memory and serial search scheme (Kolodner, 1993). The major drawback of serial search is due to the fact that each case must be examined. Naturally, given this search scheme, search time has O(n), increasing proportionally as the size of the case base increases. One advantage to serial search is that, since each case in the case base is inspected and compared to the new case, it is guaranteed that, given sound matching algorithms, the best match or matches will always be considered. Additionally, the flat memory organization allows for new cases to be quickly added to the case base since they may simply be appended to the existing list.

Hierarchical Organization of Cases: Shared Feature Networks

The idea behind hierarchical organization of the case library is to avoid the problem of large search times in systems with large case bases. This type of organization attempts to only look at a subset of the library that is the most relevant. Shared feature networks work by clustering cases together which are similar to each other. Here, it must be determined which cluster of cases is most similar to the new case and then only the cases within that cluster must be examined further for the ranking process. This clustering may be achieved by several methods (Fisher, 1987; Michalski and Stepp, 1983; Cheeseman et al., 1988; Quinlan, 1986).

As mentioned, the major advantage of using the hierarchical organization of a shared feature network is the increased efficiency of retrieval. This efficiency comes from only inspecting the most relevant subset of the case library, which is found via a breadth-first search of the network. Shared feature networks also have several disadvantages. Since not only the case descriptions must be stored for the case library, but also the network structure, the memory required to store the library is significantly larger than with a flat memory structure. Adding cases to this hierarchical structure may also cause some difficulties. Systems with flat memory storage simply append new cases to the end of the list, but adding cases to the shared feature network requires first discovering in which cluster the new case belongs by examining its description. Also, as the case base grows larger and contains a larger variety of case descriptions, the network structure may no longer be optimal for retrieval. If this becomes the case the network may need to be rebuilt with different clusters than those with which it started.

Hierarchical Organization of Cases: Discrimination Networks

Another hierarchical organization of a case base is called a discrimination network. Whereas shared feature networks focus on clustering and case discrimination is a side effect, discrimination networks focus on discriminating cases based on their important features. This is done by creating a graph with each node being a question that subdivides the set of items stored beneath it and each child node representing a different answer to the question above it (Kolodner, 1993). Each of those subsets may then be further divided by further questions, creating the network's hierarchy. Ideally, when looking at the graph as a tree, the questions nearer to the root should discriminate on more important features in order to retrieve cases that match on more meaningful dimensions. There are algorithms such as ID3 and C4.5 for building decision trees that can function as discrimination networks by reduction of entropy (Quinlan, 1986; Quinlan, 1993).

Kolodner gives several advantages and disadvantages for discrimination networks, many of which are similar to those of shared feature networks. One advantage that the former has over the latter, however, is that the discrimination network is even more efficient due to the fact that rather than comparing sub-nodes the algorithm is only asking single questions at each step through the graph. Discrimination networks also have several additional disadvantages. When parsing the tree or graph, a wrong turn at one of the early question nodes may prevent the reasoner from reaching the correct leaf node. This is not as much of a concern when using a shared feature network since several features are involved in reasoning.

Possibly worse is the situation where incomplete data prevents the reasoner from answering a question. When this happens, the reasoner cannot continue parsing the graph, so care must be taken to ensure that cases are complete. One way of avoiding this problem is by making use of a redundant discrimination network. Redundant discrimination networks are basically several versions of the discrimination tree with various orderings of the questions.

3.2 Similarity Determination and Case Retrieval in IDSDM

In this section the details of the memory organization, similarity determination and case retrieval methods used in the IDSDM system will be discussed. First, to yield a better understanding of the system for the reader, the structure of the cases will be discussed. Recall that the knowledge encoded in a case consists of a problem description, a solution and the outcome of applying the solution to the problem. Designing and implementing the case structure, which was done in collaboration with fellow graduate student Eric Flowers, is one of the contributions of this thesis.

3.2.1 Case Structure in IDSDM

As the IDSDM system is written in the Java programming language, the cases are implemented using the Java class. The cases themselves are hierarchical in nature, consisting of one main class which contains several sub-classes containing the details of the case. The hierarchical structure is shown in Figure 3.1. Each case in

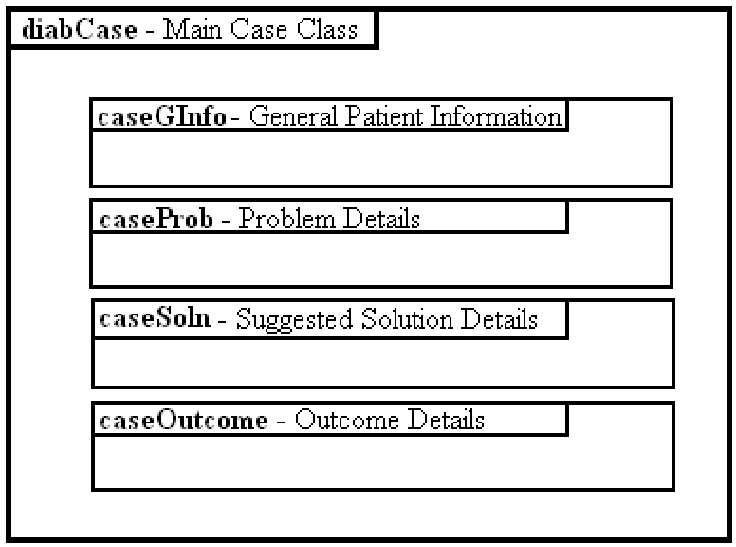

Figure 3.1: Case structure hierarchy.

the library consists of an instance of the main class, **diabCase**. Each instance of diabCase in turn consists of an instance of each of the sub-class types: **caseGInfo**, **caseProb**, **caseSoln** and **caseOutcome**. The caseGInfo class contains general information mainly regarding patient details which could affect the case such as gender, age and years with diabetes. The caseProb class contains the details of the state of the world during the time of the problem. These details include such information as blood glucose values, any food that may have been eaten, time of day that the problem occurred, whether the patient was exercising, stressors, any corrective action that may have been taken and more. The caseSoln class contains details of the solution suggested by the physician to correct the particular problem as described by the caseProb. These details include information regarding the patient's physiological condition (insulin sensitivity, carbohydrate ratio, high and low blood glucose targets, etc.) as well as the action to be taken to correct the problem (changing basal rate, suspending the insulin pump, changing carbohydrate ratio or insulin sensitivity, etc.). The CaseOutcome class contains information about the success of the suggested solution as well as information about how well and to which aspects of the suggested solution the patient adhered. These last two classes mentioned are actually not single instances within the diabCase class but, rather, arrays of the classes to allow for the description of multiple or successive solution suggestions if need be.

In total, there are 143 fields among the aforementioned classes used to describe the current state during the time that a diabetes related problem occurs. Values are stored in these fields for each existing case in the case library. When the system is given a new problem to solve, a new case is built. The solution and outcome are not initially known. As such, only the first 72 fields containing general information and the problem description are used to describe a new case. As will be discussed shortly, only a subset of the most important features from those fields will be used in similarity determination.

3.2.2 Memory Organization in IDSDM

The IDSDM system uses the flat memory organization with serial search as described in section 3.1.2 for several reasons. During initial phases of the study, hierarchical organization using discrimination networks was considered; however, the discrimination trees and rules generated by C4.5 software (Quinlan, 1993; Quinlan, 1996) and manual discrimination tree building did not yield meaningful results. Further consideration also showed that the size of the IDSDM case library did not warrant the use of hierarchical memory organization for reasons of efficiency.

At the conclusion of the Institutional Review Board (IRB) approved study of twenty diabetic patients, the system contains fifty cases which are individually stored in the structures discussed above (Marling et al., 2007a; Marling et al., 2007b; Marling et al., 2007c). As mentioned in the above discussion of flat memory organization, the time for this search is O(n) or, for IDSDM, 50 iterations of the comparison function. With this small of a case library, the time taken to search the library and compare each case to the new one is not large enough to justify the hierarchical organization of the case base.

Even so, should more cases be added to the library in the future, flat memory organization with serial search would still be a suitable configuration for IDSDM. As will be discussed in the next section, the matching algorithm is efficient enough that the case library would have to grow quite large to justify the use of a more efficient memory and search architecture.

3.2.3 Similarity Determination in IDSDM

Kolodner states that when a case-based reasoner is responsible for achieving only one kind of reasoning goal, as IDSDM is, the computation of degree of match can be a relatively straightforward process using a numerical evaluation function (Kolodner, 1993). This numerical evaluation function used for similarity determination in IDSDM is called the nearest neighbor and was briefly described in section 2.3.3. The nearest neighbor algorithm is described by Russell and Norvig as a nonparametric instance-based learning method which is simple to implement, requires little tuning and often performs quite well (Russell and Norvig, 2003).

The way that the nearest neighbor works is fairly simple and the algorithm is described by Kolodner as follows:

> "For each feature in the input case,
>
> - Find the corresponding feature in the stored case
> - Compare the two values to each other and compute the degree of match
> - Multiply by a coefficient representing the importance of the feature to the match.
>
> Add the results to derive an *aggregate match score*.
>
> This number represents the aggregate degree of match of the old case to the input. A case can be chosen by choosing the item with the largest score."

The particular function used to compute the score for each case is adapted from Kolodner and is as follows:

$$\frac{\sum_{i=1}^{n}(weight_i * similarity(oldCase_i, newCase_i))}{\sum_{i=1}^{n} weight_i - subVal}$$

where n is the number of features, $weight_i$ is the importance of the i^{th} feature and similarity is the domain dependent function comparing the i^{th} feature of old and new cases. The subVal is a value to account for irrelevant data for some types of problems. Cases in IDSDM are often incomplete, sometimes due to missing data and at other times due to irrelevance of a feature (for example, high blood glucose target during a hypoglycemic event). In the former case, if the feature is relevant to the comparison but the data is unknown, the cases are compared as usual with the similarity for that feature being zero. In the latter case, the case feature in question is not applicable to the comparison and should not be scored as usual. In this case, the similarity for the feature in question is still scored as zero, but this should not affect the case comparison and so that feature's weight is removed from the total possible weight.

Feature importance weights

Weights for each feature are in a range of 0.00 to 1.00 and are static throughout execution of the program. These weights indicate the importance of a match on a particular feature of a case with higher weights indicating greater importance. The weights were initially chosen in a manner which seemed to appropriately capture the importance of factors based on the researcher's domain knowledge and were modified after empirical testing.

Similarity functions and scores

There are 19 similarity functions called for each case comparison (see Figure 3.2). Most of them are direct comparisons of features between the newly input case and a case in the library. Others, however, are not direct feature comparisons but slightly

Category	Function	Usage
Problem Info	compareProblemType	Compares problem type
	comparePattern	Compares pattern type
	compareSitAsses	Compares situation assessment results
Hypo Details	compareRapidDrop	Compares if glucose levels fell rapidly
	compareAwareness	Compares if patients were aware of hypo event
	compareConsumption	Compares use of carbs to correct hypo event
	compareSuspension	Compares suspension of pump to correct hypo event
Hyper Details	compareRapidRise	Compares if glucose levels rose rapidly
	compareExtHigh	Compares if glucose levels were extremely high
	compareBolus	Compares use of bolus to correct hyper event
	compareInfSetChange	Compares changing of infusion set
Other Factors	CompareRelToBolus	Compares relation to bolus administration
	compareRelToDOW	Compares relation to day of week
	compareRelToTOD	Compares relation to time of day
	compareRelToExer	Compares relation to exercise
	compareTempBasal	Compares use of temporary basal rate
	compareRelToMeal	Compares relation to a meal
	compareRelToFood	Compares relation to a particular food
	compareStressors	Compares presence of stressful factors

Figure 3.2: Comparison functions used in the similarity metric.

more complicated in that they may either internally compare several other features of the case or use values derived from other factors in the case description. The result of each similarity function is a score indicating how well the corresponding features of the two cases being compared match each other. The score represents the degree of match between the features as a decimal in the range 0.00 to 1.00. Higher values indicate closer matches.

Not all 19 similarity functions are computed for each case that is inspected by IDSDM. In order to cut down on unnecessary computation time, if a similarity function is determined to be of no value to the comparison of cases its execution is omitted. An example of this in IDSDM is for the similarity functions used to compare case features unique to hypoglycemic episodes. If the problem is not related to hyperglycemia, then the similarity determination functions for features related to hyperglycemia are not applicable and their execution is omitted.

The overall execution of the similarity determination module may be seen in the pseudo-code presented in figures 3.3 and 3.4. The code begins by first determining, on a very abstract level, if the cases are similar enough to continue comparing. If they are similar enough to continue comparison, IDSDM compares the cases on four different categories: general problem details, hypoglycemic details, hyperglycemic details, and details regarding the case's relationship to other various factors.

Individual feature comparisons and score determinations are each completed in their own java method. These comparisons are based on the Boolean or enumerated type values of the features in the existing case and the new case. Boolean comparisons include the rapid glucose level drop, corrective consumption and pump suspension comparisons from the hypoglycemia details section, the rapid glucose level rise, extreme high, corrective bolus and infusion set change comparisons from the hyperglycemia details section, and the related to bolus, related to exercise, temporary basal rate, related to specific food and related to stressful factors comparisons from

- Determine match score of problem types.
  - If problem type match scores are below the threshold for further comparison, no further comparisons are performed and the case's score is computed.
  - If problem type match score is above the threshold for further comparison, continue comparing case features.
- Compare general problem details as follows:
  - Determine match score for situation assessment and add it to the aggregate score.
  - Determine match score for problem pattern and add it to the aggregate score
- If problem type does not concern hypoglycemia, add hypoglycemia detail factor weights to aggregate subtracted score and continue to next step. If problem type concerns hypoglycemia, compare hypoglycemia details as follows:
  - Determine match score for rapid decrease in glucose level and add it to the aggregate score.
  - Determine match score for patient awareness of hypoglycemia and add it to the aggregate score.
  - Determine match score for corrective consumption and add it to the aggregate score.
  - Determine match score for insulin pump suspension and add it to the aggregate score.
- If problem type does not concern hyperglycemia, add hyperglycemia detail factor weights to aggregate subtracted score and continue to next step. If problem type concerns hyperglycemia, compare hyperglycemia details as follows:
  - Determine match score for rapid increase in glucose level and add it to the aggregate score.
  - Determine match score for extremely high glucose level and add it to the aggregate score.
  - Determine match score for corrective insulin administration and add it to the aggregate score.
  - Determine match score for infusion set change and add it to the aggregate score.

Figure 3.3: Pseudo-code representation of IDSDM similarity determination module (Part 1 of 2).

- Compare details regarding the cases relationship to other various factors as follows:
  - Determine match score for relationship to bolus administration and add it to the aggregate score.
  - Determine match score for relationship to day of week and add it to the aggregate score.
  - Determine match score for relationship to time of day and add it to the aggregate score.
  - Determine match score for relationship to exercise and add it to the aggregate score.
  - Determine match score for temporary basal rate and add it to the aggregate score.
  - Determine match score for relationship to meal and add it to the aggregate score.
  - Determine match score for relationship to specific food and add it to the aggregate score.
  - Determine match score for relationship to stress factors and add it to the aggregate score.
- Determine case match score as follows:
  - Determine total possible weight by subtracting the aggregate subtracted score from the total importance weight of all factors.
  - Divide the aggregate score by the total weight.
- Assign score to case.

Figure 3.4: Pseudo-code representation of IDSDM similarity determination module (Part 2 of 2).

the various features section. Pseudo-code for the general operation of these types of comparisons can be seen in figure 3.5. Enumerated type comparisons include the problem type, pattern and situation assessment comparisons from the general problem details section and the patient awareness comparison from the hypoglycemia details section. Pseudo-code for the general operation of these types of comparisons can be seen in figure 3.6. The related to day of week, related to time of day and related to meal comparisons from the various features section make use of a combination of Boolean and enumerated type values for comparison. These methods base the decision of whether to compare the enumerated type values of a feature on the Boolean values of another feature.

- Determine the degree of match as follows:
  - If the value of the feature from the existing case and the value of the feature from the new case are either both true or both false, the degree of match is 1.0.
  - If the value of the feature from one case is true while the value of the feature from the other case is false, the degree of match is 0.0.
- Determine the feature's match score as follows:
  - Multiply the feature's degree of match by the importance weight of the feature.

Figure 3.5: Pseudo-code representation of Boolean valued features.

- Determine the degree of match as follows:
  - Assign a value in the range of 0.0 to 1.0 based on similarity tables which contain the degree of match values for all combinations of enumerated type values from the existing case and the new case.
- Determine the feature's match score as follows:
  - Multiply the feature's degree of match by the feature importance weight.

Figure 3.6: Pseudo-code representation of comparison of enumerated type valued features.

3.2.4 Case Retrieval in IDSDM

IDSDM uses the two-step retrieval process described in section 3.1.1. The first step is used to narrow down the case base to a smaller subset which contains only cases which are similar enough to inspect more closely. This is accomplished by comparing only the problem type of the case in question with that of those in the case library using the "problem type" comparison function. If the case is determined not to be similar enough to continue, no more comparison functions are executed. If, based on problem type, the cases are similar enough to warrant further examination, the remaining comparison methods are executed as described in the "Similarity functions and scores" discussion in section 3.2.3 above.

After the newly input case has been compared against each case and match scores have been assigned, IDSDM must decide which, if any, cases will be returned. Recall that the idea behind IDSDM is to, when a new problem is encountered, analyze blood glucose and lifestyle data and to recommend solutions to individual problems in blood glucose control (Marling et al., 2007b). The solutions that are recommended for the new problem are taken, either directly or after some modification, from the returned cases. As such, care must be taken to only return good, relevant cases. Once all cases in the library are scored there are several methods of choosing which to return as good matches. These methods include returning all cases whose score is above a certain threshold value and returning the top K cases where K is a fixed number (K-nearest neighbor) (Russell and Norvig, 2003).

IDSDM combines both of the above methods for determining which cases to return. The top scoring cases are those that the matching algorithm has determined are the most similar, but it should also be ensured that the returned cases are relevant. Blindly returning the K highest scoring cases allows for the possibility of returning poorly matching cases when there few or no high scoring cases, as could be the situation when the newly encountered problem is different from anything previously seen. On the other hand, returning all cases above a certain threshold score could result in a very large set of returned cases, making it difficult to determine what solution to apply to the new case. Therefore IDSDM will only return the top matching case if the match score assigned to it is above a 0.78 threshold score. If no cases are above that threshold, the system will report that no relevant cases were found. This threshold value was chosen based on domain knowledge and initial testing results of the system.

3.3 Putting It All Together

The previous sections have described the individual pieces of the similarity determination and case retrieval functionality. An example execution of the IDSDM system will now be given to demonstrate that functionality.

3.3.1 Similarity Determination Example

Consider the case presented in Figure 3.7. This case will act as the newly input case for this example execution of the similarity determination metric for IDSDM. As indicated in the figure, this problem involves the patient experiencing overnight hypoglycemia. This is Case 14 from the case library. The solution for this problem suggested by the physician after reviewing the case details was for the patient to lower

The problem is described as: Patient has nocturnal lows.
Instructions for the patient are: Lower the early morning basal rate from 0.8 to 0.7 between midnight and 7AM. Always have a bedtime snack.

Figure 3.7: Brief description of input case.

their overnight basal rate and to have a bedtime snack. The first part of this solution will cause the patient to be receiving less insulin during those hours, which will cause the glucose in their system to remain in their body longer and hopefully prevent their glucose levels from falling to hypoglycemic levels. The second part of the advice, consumption of a bedtime snack, will introduce carbohydrate into the body to allow the patient to begin the night with a slightly increased blood glucose level, giving them farther to fall during the night before reaching hypoglycemic levels.

Once the similarity determination metric is activated, it proceeds through the case base in order, assigning scores to each case which represent that case's similarity to the input case. The first thing that occurs for each case comparison is for the total possible weight of all features to be determined. This weight, the combined feature weight of all factors in the general problem details, hypoglycemia details, hyperglycemia details and relationship details sections, totals to 3.95.

The next step is to compare the cases based on problem type alone. The importance weight of this feature is 0.85. This comparison determines if further consideration is necessary. The first case encountered in the case library is presented in Figure 3.8. As can be seen in the figure, this case deals with hyperglycemia whereas the input case deals with hypoglycemia. As such, the similarity between the two problem types

The problem is described as: Patient is consistently a little high after breakfast.
Instructions for the patient are: Change the breakfast carb ratio from 1:13 to 1:12.

Figure 3.8: Brief description of Case 1.

is taken from the similarity table as 0.00. When multiplied by the feature weight, 0.85, this feature's contribution to the overall score is 0.00, well below the threshold for further evaluation. Since the two cases are not determined to be similar enough to continue comparisons, the score of 0.00 is assigned to Case 1 and IDSDM moves on to the next case comparison. These comparisons continue on through the rest of the cases in the case library.

The next case of interest is Case 49, presented in Figure 3.9. Again, this case deals with overnight hypoglycemia. The physician recommended solution to this problem is to have a bedtime snack which, again, allows the patient to begin the night with a slightly elevated blood glucose level and thus more room to fall before reaching hypoglycemic levels.

The problem is described as: Patient has overnight low.
Instructions for the patient are: Have at least a small bedtime snack, perhaps a glass of milk.

Figure 3.9: Brief description of Case 49.

IDSDM again begins comparison of these cases based solely on the problem type feature. As both cases deal with hypoglycemia, the similarity match is 1.00. When multiplied by the feature importance weight, the contribution to the total score is 0.85 out of 0.85, a full match. This score is sufficient to continue comparison, so IDSDM moves on to comparing the remaining general problem detail features. As the input case was determined by the situation assessment module as early morning CGMS lows and Case 49 was not found by the situation assessment module, this feature matches 0.00, which contributes none of the 0.20 importance weight of this feature to the score. The pattern features do not match, as the input case occurred several days in a row while the problem described in the existing case occurred sporadically, yielding a similarity of 0.00 and contributing none of the 0.20 importance weight to the score. At this point, the score is 0.85 out of the possible 1.25 weight for the general problem details portion of comparison.

Next, the hypoglycemia details of the two cases are compared. All features in this section receive a similarity score of 1.00:

- Neither problem involves the rapid rise of glucose levels.
- The patient is aware of the hypoglycemia in both cases.
- The patient attempted to correct the hypoglycemia by consuming carbohydrates in both cases.
- The patient did not suspended their insulin pump in either case.

These similarities yield the contribution of the entire 0.80 importance weight of the hypoglycemia details portion of comparison. At this point, the score is now 1.65 out of a possible 2.05 weight for all features compared thus far.

The next portion of comparison is that for hyperglycemia details. As the input case does not involve hyperglycemia, these features, rapid rise, extreme high blood glucose levels, corrective bolus and infusion set change, are not relevant to comparison. These features contribution to the total score is 0.00 and the irrelevant feature weight, subVal, as described in section 3.2.3, is increased to the combined weight of these features, 0.80.

Finally, the remaining features are compared. All features of the input case and case 49 receive a similarity score of 1.00:

- Neither problem is related to the administration of a bolus dose of insulin.
- Neither problem is related to the day of the week.
- Both problems are related to the time of day; specifically the early morning hours.
- Neither problem is related to exercise.
- Neither problem involved the setting of a temporary basal rate.
- Neither problem was related to the consumption of a meal.
- Neither problem was related to a specific food.
- Neither problem was reported as having occurred during a stressful event.

This contributes the combined value of the importance weights from these features, 1.10, to the total score. The irrelevant feature weight remains at 0.80.

After all of the above comparisons are completed between these two cases, case 49 is left with a total score of 2.75 out of a possible 3.95. After the weight of the irrelevant features is subtracted, the total possible weight becomes 3.15. The scoring of 2.75 out of a possible 3.15 gives case 49 a 0.87 similarity score with respect to the input case. This was the highest scoring case from the case library and is the case returned by the IDSDM system. The evaluating physicians all indicated that the problem similarity was "very similar" and two out of three indicated that the returned solution was "very beneficial" with the third indicating "somewhat beneficial."

3.3.2 Similarity Determination as Part of the Overall IDSDM Operation

Figure 3.10 provides a software level overview of the IDSDM system as a whole, including both finished and unfinished modules. Software developed for this research resides in the components labeled Similarity Determination, Scores and Relevant Case(s) on the IDSDM system overview shown in Figure 3.10. The computer terminal element labeled **Patient** in the upper-left of the figure signifies the method of input for patient data during the initial study. This represents the patient entering data as described in section 2.3.2. The **Website** element represents the website with which the patient interacts to enter data, as shown in Figure 2.3. This patient data is then stored in the **Database**. The **Problem Detector** and **New Case Finder**(unfinished) elements then work together to scan the patient data for poten-

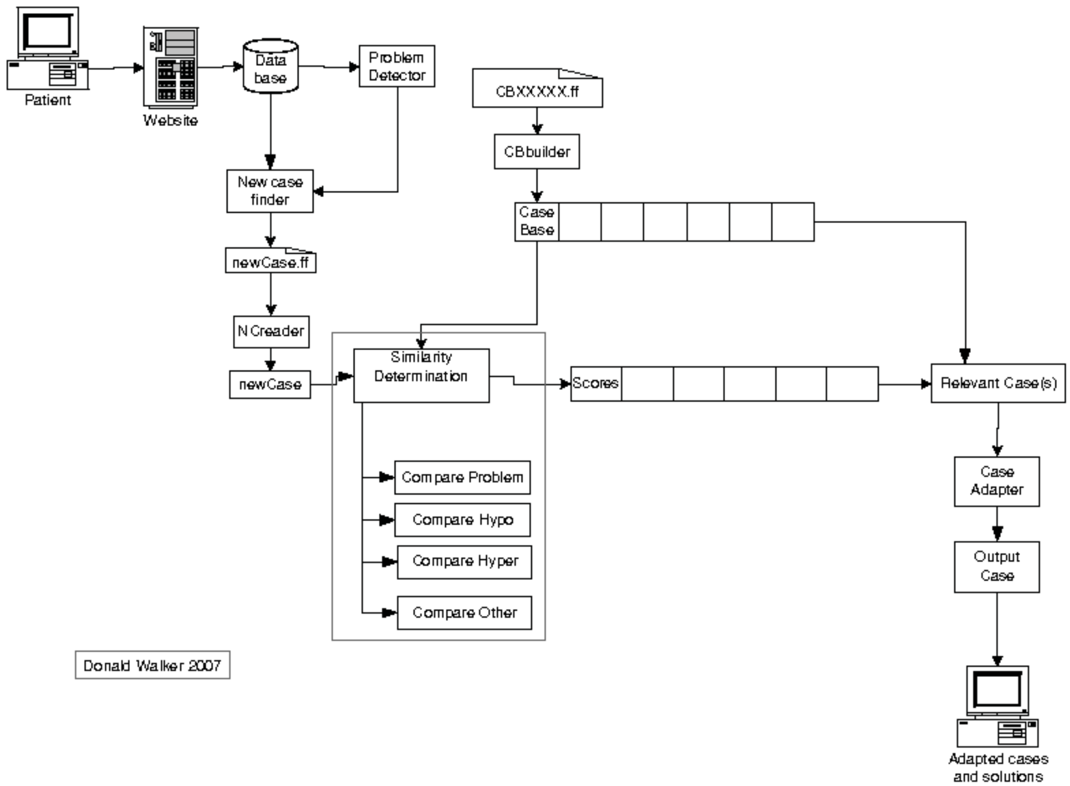

Figure 3.10: Software level overview of IDSDM.

tial glucose management problems and create a text file (**newCase.ff**) containing the details surrounding the problem. This file is in turn read by the **NCreader** function and turned into **newCase**, an instance of diabCase as described earlier. The **CBXXXX.ff** element represents the file containing the case base. This file is read by the **CBbuilder** function and turned into an array of type diabCase, represented by the element **Case Base**.

Once this point in execution is reached, it is time to execute the portion of the system that is the focus of this research. The **Similarity Determination** function, labeled *Similarity Determination* in figure 3.10, executes the similarity determination code as described above to create the **Scores** array which contains the calculated similarity between each case in the library and the newly input case. Once all scores are calculated, the **Relevant Case(s)** are retrieved as per the requirements stated above. At this point, the returned cases are passed on to the **Case Adapter**(unfinished) which, if needed, modifies any parameters of the case to tailor it to the specific needs of the patient whom the new case is regarding. Once the case is either modified or determined to not require modification, it is sent to the **Output Case** function which displays the suggested solution to the newly encountered problem, as represented by the **Adapted Cases and Solutions** element.

Chapter 4

EVALUATION AND FUTURE WORK

This chapter explains the evaluation methodology and provides data pertaining to the performance of the similarity metric for the IDSDM system based on multiple testing and evaluation procedures. It also offers suggestions for what the next steps in development for IDSDM might be.

4.1 Evaluation of Matching and Retrieval Metric

Evaluation of the similarity metric for IDSDM was done in two main phases. The initial testing phase was subdivided into two main procedures. The first of these procedures involved performance review by the researchers and faculty advisor of the project while the second involved review by three physicians with formal domain specific knowledge. The second phase of testing again involved those involved in diabetes care practice and was performed after implementing improvements to correct issues found during initial testing.

4.1.1 Initial Testing

Preliminary Evaluation Methodology

Initial testing of the code was accomplished by performing what is called "leave one out" testing. Evaluation is done by comparing each of the 50 cases in the case base to all other cases. This was accomplished by first creating a file containing only the problem portion of each case with a file name of the form "newCase" followed by the case number followed by ".ff" and placing these files in their own directory. During full system execution, the case to be compared against the case base is automatically generated from live patient data and saved in a file called "newCase.ff" which is then read into memory to be compared to the existing cases to determine the degree of match. To simulate this process during testing, a script was created which copied each of the manually created problem test files into the working directory one by one and called the comparison code. This process simulates that of the automatic problem detector finding 50 new problems, compiling the necessary data, generating the test file and running the comparison metric.

This type of testing is useful as a test of the quality of the comparison and retrieval metric because it is known what solution the physician recommended for each specific problem. With this information, along with the comparison results, it can be observed how well the problem descriptions of the two cases match, as well as the advice that would be given to a patient with such a problem. Ideally, similar problems should have similar solutions. By looking at the suggested advice for the highest matching returned case(s) and comparing them to the actual advice given for the test problem, it can be determined if that is actually the case.

AI researchers working on the IDSDM project, including the faculty advisor, reviewed the results of all 50 test runs to determine the similarity of returned cases. These test results appeared promising in that the problem descriptions of highly matching cases seemed similar to the case being tested while the problem descriptions for lower scoring cases did not. However, none of the AI researchers nor the faculty advisor is a physician, so further evaluation was performed by physicians who treat patients with diabetes.

For the second portion of the preliminary testing of IDSDM, a subset of twelve cases was randomly chosen for review by three physicians specializing in diabetes care. Each physician was given a packet containing a form for each of the twelve cases. Each form shows the brief description of the original problem and the physician recommended solution to the problem along with the match scores, brief problem descriptions, and physician recommended solutions for the top three matching cases. An example of the form can be seen in figures 4.1 and 4.2. In addition to this material, each physician also received a packet containing the complete human generated and software generated reports for each relevant case.

Preliminary Evaluation Results

When examining the following results, it is important to bear in mind that for the preliminary evaluation the top three matching cases were returned regardless of their match score. As such, it is possible for a case with no similar cases to have very low scoring cases returned as the best matching cases. This inclusion of poorly matching cases was deliberate as it helps to determine an appropriate threshold score below which no cases will be returned. During normal operation, it is possible for a new

Case number 1
Problem Description: **Patient is consistently a little high after breakfast.**
Solution: Change the breakfast carb ratio from 1:13 to 1:12

Best Matching Cases:
Case 44: 0.93
Problem Description: Patient has post-breakfast highs.
Solution: Change the carb ratio with breakfast from 1:15 to 1:12.

Case 26: 0.91
Problem Description: Patient has post-breakfast highs.
Solution: Use a dual-wave bolus, instead of a normal bolus, with breakfast.

Case 6: 0.88
Problem Description: Patient's blood glucose creeps up in the early morning and spikes after breakfast.
Solution: Increase the early morning basal rate from 0.3 to 0.35 between 5AM and 7AM

Please circle your response to each of the following statements:

For the best matching case:

1) The problems in the original case and the matching case are:
   1. Very Similar
   2. Somewhat Similar
   3. Somewhat Dissimilar
   4. Very Dissimilar

2) Applying the matching case's solution to the original problem would be:
   1. Very Beneficial
   2. Somewhat Beneficial
   3. Neither Beneficial nor Detrimental
   4. Somewhat Detrimental
   5. Very Detrimental

Figure 4.1: Sample evaluation form, page 1

problem to arise that is truly different from those contained in the case base. When this occurs, the system must recognize that it has no useful solution to offer.

In this phase of testing, 62% of responses indicated that the best matching case was very similar or somewhat similar to the original case, as shown in Figure 4.3. The second and third best matching cases were indicated to have some degree of similarity by 50% and 34% of responses, respectively. The first matching case's solution was indicated by 67% of responses to have some degree of benefit and by another 25% to be of neutral benefit as shown in figure 4.4. The solutions for the second and third

```
For the second best matching case:

 3) The problems in the original case and the matching case are:
       1. Very Similar
       2. Somewhat Similar
       3. Somewhat Dissimilar
       4. Very Dissimilar

 4) Applying the matching case's solution to the original problem would be:
       1. Very Beneficial
       2. Somewhat Beneficial
       3. Neither Beneficial nor Detrimental
       4. Somewhat Detrimental
       5. Very Detrimental


For the third best matching case:

 5) The problems in the original case and the matching case are:
       1. Very Similar
       2. Somewhat Similar
       3. Somewhat Dissimilar
       4. Very Dissimilar

 6) Applying the matching case's solution to the original problem would be:
       1. Very Beneficial
       2. Somewhat Beneficial
       3. Neither Beneficial nor Detrimental
       4. Somewhat Detrimental
       5. Very Detrimental



  7) Which of the three matching cases do you think is most similar
       to the original case?
       1. the first
       2. the second
       3. the third
       4. None of them is really very similar to the original case
       5. All of them are equally similar to the original case
```

Figure 4.2: Sample evaluation form, page 2

best matching cases were indicated by 52% and 41% of responses, respectively, to have some degree of benefit. It may also be noted that the solutions for the top three best matching cases were indicated by 8%, 11% and 19% of responses, respectively, to have any degree of detriment and none were indicated to be “very detrimental.”

Figure 4.5 depicts the physicians’ opinions of which returned case most closely resembles the original input.

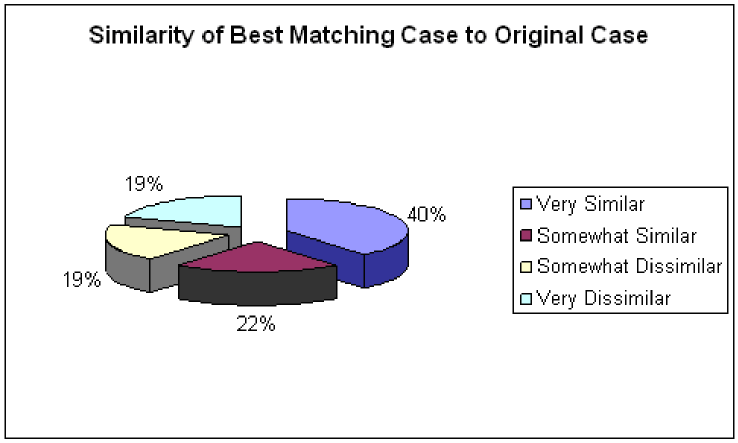

Figure 4.3: Similarity of best matching case to input case.

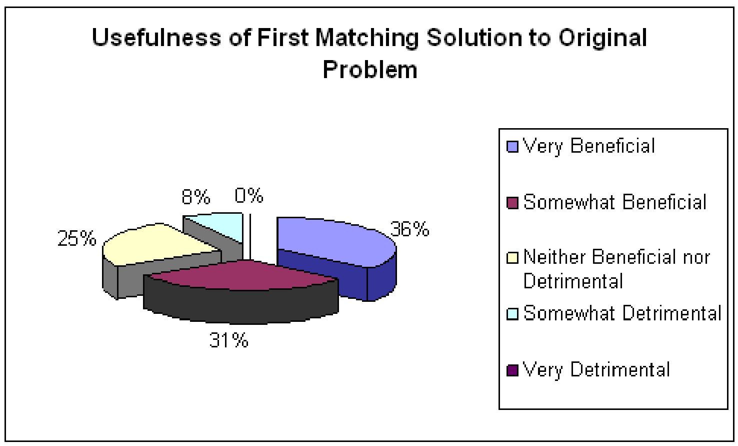

Figure 4.4: Usefulness of best matching solution to the input problem.

The above results indicated that the similarity metric was generally scoring similar cases higher than dissimilar ones. However, they also indicated that there was room for improvement.

The major improvement that was made to the similarity determination algorithm was the decision to use a two-step retrieval process rather than a one-step retrieval process. Prior to this evaluation, all included features of all cases in the case base

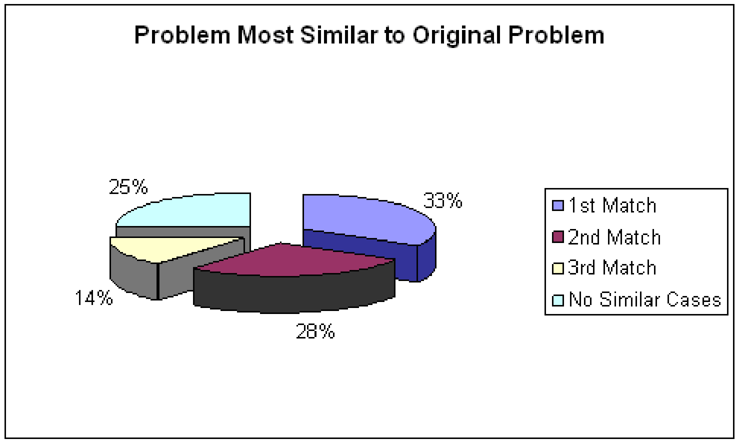

Figure 4.5: Matching case most similar to input case.

were compared to those of the new case. During this testing, it was observed that cases that were not very similar were being given relatively high scores. These scores were not as high as those of the similar cases, but they did not adequately represent the dissimilarity between such cases and the newly input case. It was also observed that the scores of all cases were clustered together in the medium to high score range. Changes were made to the similarity determination metric such that instead of comparing all included features of all cases, an initial comparison is done between the newly input case and the case from the case library based solely on the problem type value. If, based on this feature alone, the cases are determined to be "similar enough," comparison of other features continues as usual. Otherwise, all other comparisons are skipped resulting in a very low score for dissimilar cases.

Another improvement made to the similarity determination algorithm following this initial evaluation was to increase the ratio of the importance values of the problem type, situation assessment, related-to-bolus and related-to-exercise comparisons to the importance values of the other factor comparisons. This change resulted in the comparisons being based primarily on the most important features.

This testing also helped to determine an appropriate score threshold for returning cases. The threshold score value is used as a cut-off point above which cases will be returned as similar to the input case. IDSDM will not return any cases scoring below this value. As can be seen in figure 4.6, cases scoring within the 0.81 to 1.00 range are among the most similar cases. Solutions for the returned cases scoring in the 0.61 to 0.70 range are not indicated as having any benefit for the original case and should therefore not be returned as good results. Cases scoring in the 0.71 to 0.80 range are indicated as having a larger percentage of solutions that are only somewhat beneficial or of neutral benefit. This indicates that the threshold value likely should be somewhere toward the high end of this range.

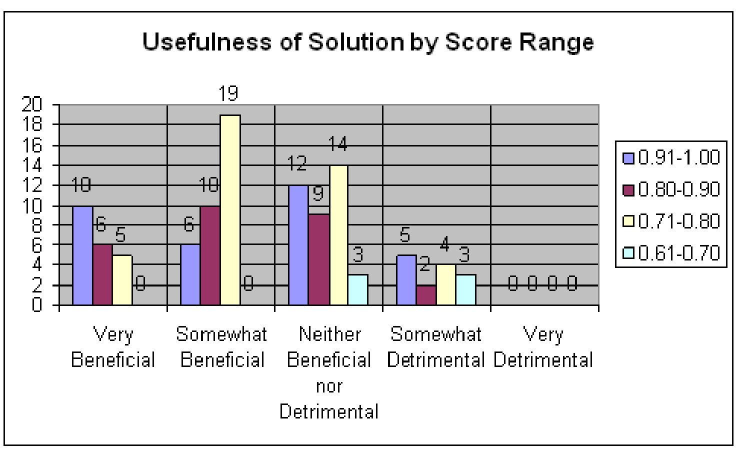

Figure 4.6: Usefulness of returned solution by score range.

4.1.2 Phase Two of Testing

Second Phase Evaluation Methodology

The second phase of performance testing for the similarity metric of the IDSDM system is similar to that of the initial test in that it utilizes leave one out methods. Twenty percent of the cases from the existing case library were chosen randomly to be compared against the remaining cases. The format of the evaluation changed slightly from that of the initial evaluation in that the single most highly matching case was returned only if its match score was above the threshold value. The threshold value for returning cases was 0.79 for this phase of testing. However, all of the test cases returned matches that were well above this threshold indicating that, in theory, all of the test cases had cases in the case library that were "similar enough" to be returned.

Again, the same three physicians specializing in diabetes care were asked to evaluate the degree of match between the input case and the returned case as well as the benefit of applying the solution of the matching case to the input case. Each physician was given a packet with an evaluation form for each case, an example of which may be seen in figure 4.7.

While opportunity for improvement on the similarity metric is still present, progress since the initial evaluation can clearly be seen in the results of this second phase of testing. As can be seen in figure 4.8, 80% of the responses indicate that the problem description of the returned case was either "very similar" or "somewhat similar" to the problem description of the original case. This is a vast improvement over the 62%

Test Case 4 (14)

The problem is described as: Patient has nocturnal lows.
Instructions for the patient are: Lower the early morning basal rate from 0.8 to 0.7 between midnight and 7AM. Always have a bedtime snack.

Matching Case: Case 49 (0.94)
The problem is described as: Patient has overnight low.
Instructions for the patient are: Have at least a small bedtime snack, perhaps a glass of milk.

Please circle your response to each of the following statements:

4) The problem in the original case and the problem in the matching case are:
   1. Very Similar
   2. Somewhat Similar
   3. Somewhat Dissimilar
   4. Very Dissimilar

5) Applying the matching case's solution to the original problem would be:
   1. Very Beneficial
   2. Somewhat Beneficial
   3. Neither Beneficial nor Detrimental
   4. Somewhat Detrimental
   5. Very Detrimental

Please enter any comments regarding the results of this match here:

Figure 4.7: Sample evaluation form used for the second phase of performance testing.

of responses indicating the same during the first phase of evaluation.

There was also a 6% increase from 67% to 73% of responses indicating that applying the returned solution to the input case would provide some degree of benefit to the patient, as can be seen in figure 4.9. This increase in the utility of the returned solution also resulted in fewer instances of the returned solution being judged

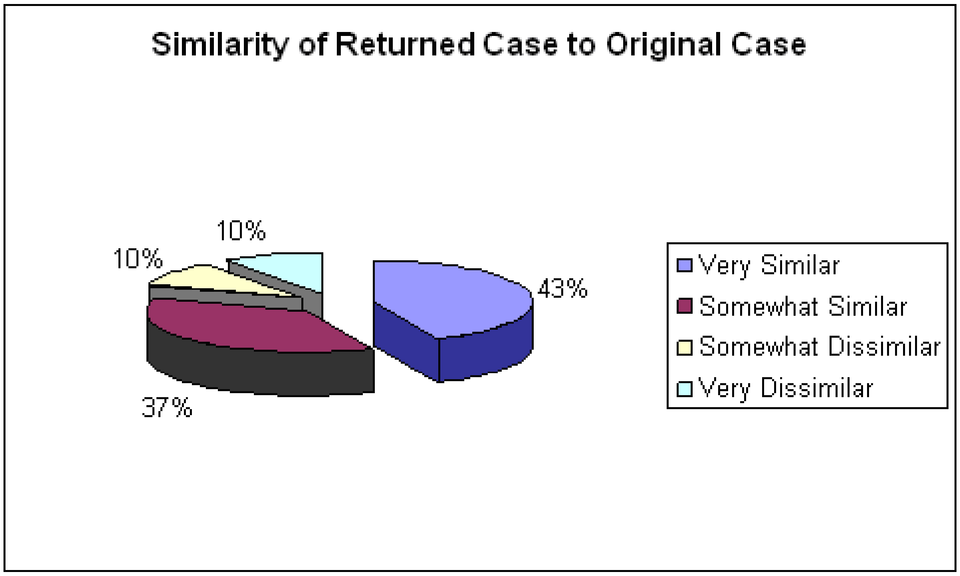

Figure 4.8: Similarity of returned case to input case.

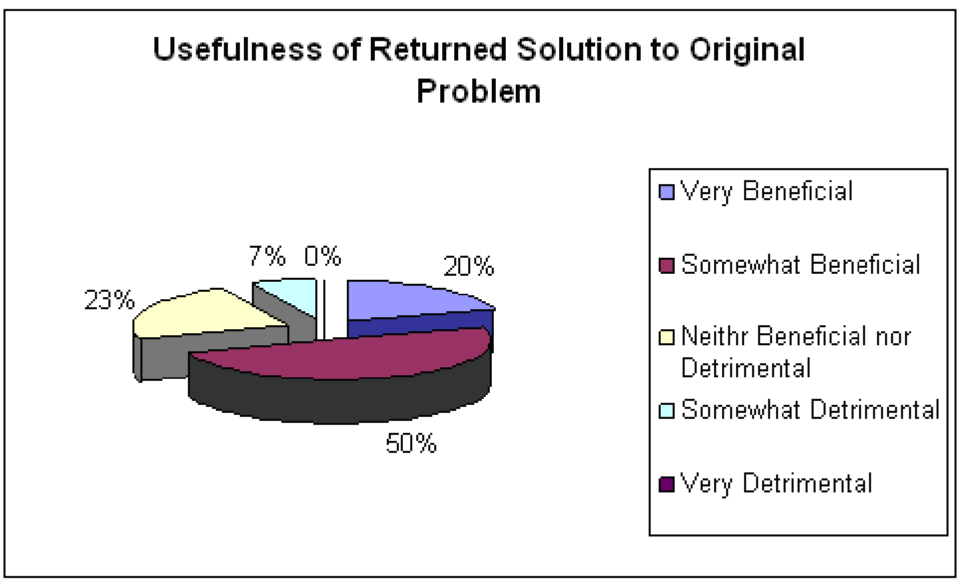

Figure 4.9: Usefulness of returned solution to input case.

as “neither beneficial nor detrimental” or placed in the already sparse category of “somewhat detrimental.”

The above gains in the performance of the IDSDM system’s similarity metric may be attributable to a combination of the migration from a single-step retrieval process to a two-step retrieval process as well as the modifications to feature weights. While improvements have been made in the short time between evaluations, the system does not provide useful solutions 100% of the time. When used in a clinical situation, a decision support system must give appropriate advice in order to avoid harm to the patient. Ideas for how to further improve IDSDM are presented in section 4.2.

There may be several explanations as to why IDSDM would return a case as being similar when it is indicated by the physician to have some degree of dissimilarity. For instance, one test case which was scored by two of the evaluating physicians as being dissimilar involves improper bolusing by the patient. In the test case, the patient would administer a large, uncalculated bolus dose of insulin when changing infusion sets without regard for his current blood glucose level which resulted in the patient often going low when changing infusion sets. The returned case had a very high match score but did not involve infusion set changes at all. This case involved a patient who would administer a pre-meal bolus dose of insulin large enough to cover the carbohydrates in the meal even when the patient’s blood glucose is low. Both of these problems resulted in hypoglycemic episodes, but for reasons that may be easily seen as dissimilar by a human. On the other hand, the circumstances of the cases both involve a hypoglycemic episode brought on by the administration of an inappropriate bolus dose of insulin. There are also many other matching factors between the cases which cause them to appear similar to IDSDM. Despite these similarities, the advice given in these two cases differs greatly because of the variation in circumstances surrounding them. Some suggestions on how to remedy these types of errors are discussed in section 4.2.

One other example of note is a case involving post-breakfast hyperglycemia. In this test case, the patient's blood glucose level was creeping up early in the morning and then spiking immediately after breakfast. The returned case involved only the post-breakfast high glucose levels. At first glance these cases may appear nearly identical; however, upon closer examination by the evaluating physicians they are actually found to be two separate problems. All physicians indicated that these problems were "somewhat similar" and two of them rated the advice given by IDSDM as "somewhat beneficial" while the other indicated that it was "somewhat detrimental."

Here, the problem in the returned matching case appears to be caused by the patient not taking enough insulin to handle the carbohydrates contained in the meal. To correct the problem the physician suggested changing the carbohydrate from one unit of insulin per 15 grams of carbohydrates to one unit per 12 grams of carbohydrates. In the original case, however, the physicians all agree that while the spike in blood glucose levels after breakfast may be due to similar causes as the returned case, it is compounded by a condition that they believe to be present called the dawn phenomenon (Stephenson and Schernthaner, 1989; Bolli et al., 1993). This condition is a result of the body's decreased need for insulin between the hours of midnight and 3 A.M. and the subsequent increase in the need for insulin between the hours of 5 A.M. and 8 A.M., which alters the body's natural insulin sensitivity. (Bolli et al., 1993). As a result of the patient's body needing more insulin just before waking, the physician recommended that the patient increase the early morning basal rate of insulin from 0.30 to 0.35 during the hours of 5 A.M. to 7 A.M. The pre-breakfast problem was addressed first, as its treatment could also affect the post-breakfast problem. This test case was the only case in the case base containing both pre- and post-breakfast problems, so a better match could not be found.

There was some degree of disagreement among the physicians on the similarity of the returned matches. For instance, some test cases involved problems described as "wide swings in glucose levels" or "bouncing between high [glucose levels] and low [glucose levels]." This lack of control over blood glucose levels could be a result of some initial problem followed by a series of overcorrections. These swings could also indicate very poor glucose control in general. IDSDM, in three test cases, matched cases with swinging glucose levels to cases where the patient overcorrected for a hyperglycemic event and experienced a hypoglycemic event. This was the one type of problem where the match received scores ranging from "very similar" to "very dissimilar." Further discussion will be needed to resolve these differences of opinion so that the similarity metric can be appropriately adjusted.

4.2 Future Work for IDSDM

4.2.1 Improving Similarity Determination and Case Retrieval Performance

The first and foremost way to improve IDSDM, as with many case-based reasoners, is to expand the size of the case library. CBR systems rely on their "memories" or "experiences" to solve the new problems which they encounter, just as humans do in everyday situations. A CBR system with a very small repertoire of past experiences in the form of cases would be like the mechanic from the example in section 2.2.1 having very limited experience with cars. Say, for instance, that the only vehicles the mechanic had ever worked on were a few semi-trucks. If someone brought a compact sports car to the garage, the mechanic might have a good enough understanding of how vehicles work in general to make some minor repairs on the sports car, but it is doubtful that he would have the appropriate knowledge to fix all problems that it might have.

The same is true for the case-based reasoning system. If the case library is too small, it may not contain cases similar enough to the input case to return an appropriate match. In general, the more cases that are in a reasoner's case library, the more likely the system is to find a case that is similar. A CBR system, however, is not guaranteed to find a match; there is always the possibility that a new case is so unique that, even with a very large case base, there are no similar cases. It follows that adding more cases to the IDSDM case library would, in many cases, improve the chances of finding a case more closely resembling the input case and thus give advice better suited for the problem at hand.

It should be noted that the increase in the quality of match with increasing size of the case library is not infinite. There is a point of diminishing returns beyond which growing the case base further does not yield any increase in performance. At some point in the lifespan of the growing case library the *utility problem* is encountered. The utility problem is encountered when "the cost associated with searching for relevant knowledge outweighs the benefit of applying this knowledge" (Smyth and Keane, 1995). Determining the utility of cases and deciding which ones to add to the library is beyond the scope of this thesis. The decisions that must be made include determining whether a new case is so similar to an existing case that it would not provide any additional advice, whether a new case is so unique and encountered so infrequently that adding it to the case library would only cause increase of computation time during search, and other similar factors affecting utility. Two ways to avoid the utility problem include only adding cases with high utility and limiting the size of the case library, beyond which cases may be trimmed to ensure high overall utility of the case base (Smyth and Keane, 1995).

Another possible way to get IDSDM to return better results may be to add more categories of problem types. Again, a compromise must be reached because adding more problem type categories could help to differentiate between different kinds of problems within each category but it could also cause the cases to become too specific to match other similar problems with potential benefit. Current problem types are hypoglycemia, hyperglycemia, potential hypoglycemia, potential hyperglycemia, hypoglycemia followed by hyperglycemia and hyperglycemia followed by hypoglycemia. One possibility for gaining the differentiation capability without becoming too specific would be to add another field such for "problem subtype" to the case description.

A third possibility for improving the quality of match is to further discriminate between cases based on patient specific data not specifically pertaining to the problem at hand. The physicians involved with this study have provided anecdotal evidence suggesting that factors such as marital status, age and other physiological factors may have an impact on how the body responds to the disease and its treatment. All of this data is readily available in the existing IDSDM patient database. For this to be of great benefit, the case base would need to be larger and contain problems, solutions and outcomes for a wide variety of patients.

Another addition to the current version of IDSDM that could improve discrimination between cases would be additional feature comparisons. For instance, two more checks that could be included are for overcorrection for hyperglycemia and overcorrection for hypoglycemia. These are fairly common problems and many additional problems stem from overcorrection in either direction. A prototype of the method to check for overcorrection for hyperglycemia is implemented in an experimental version of IDSDM, but is not ready for inclusion in the full version. Again, however, when the number of fields being compared is increased the system runs the risk of becoming too specific and missing good matches.

4.2.2 The Future of IDSDM

IDSDM modules under development and planned for future development

There are two other IDSDM modules under development by other researchers that will improve the overall functioning of the system. The first of these modules is the new case building module. This module, working in conjunction with the problem detection module already in place, begins by scanning the patient database for problems with the patient's glucose management. When it finds a problem it gathers all necessary data regarding the circumstances surrounding the problem and stores the data in a file formatted for use by the similarity determination and case retrieval module. This file is the element labeled "newCase.ff" in the upper-left of figure 4.10, which also appeared in Chapter 3. This functionality is integral to the stand-alone operation of the IDSDM system as it finds the problems that must be corrected and provides the similarity determination and case retrieval module with a starting place for solving the problem.

The next module in development for the IDSDM system is the solution adaptation module, labeled "case adapter" in the lower-right of figure 4.10. The function of this module is to make minor adjustments in the solution suggested in the case returned by the similarity determination and case retrieval module. This functionality is im-

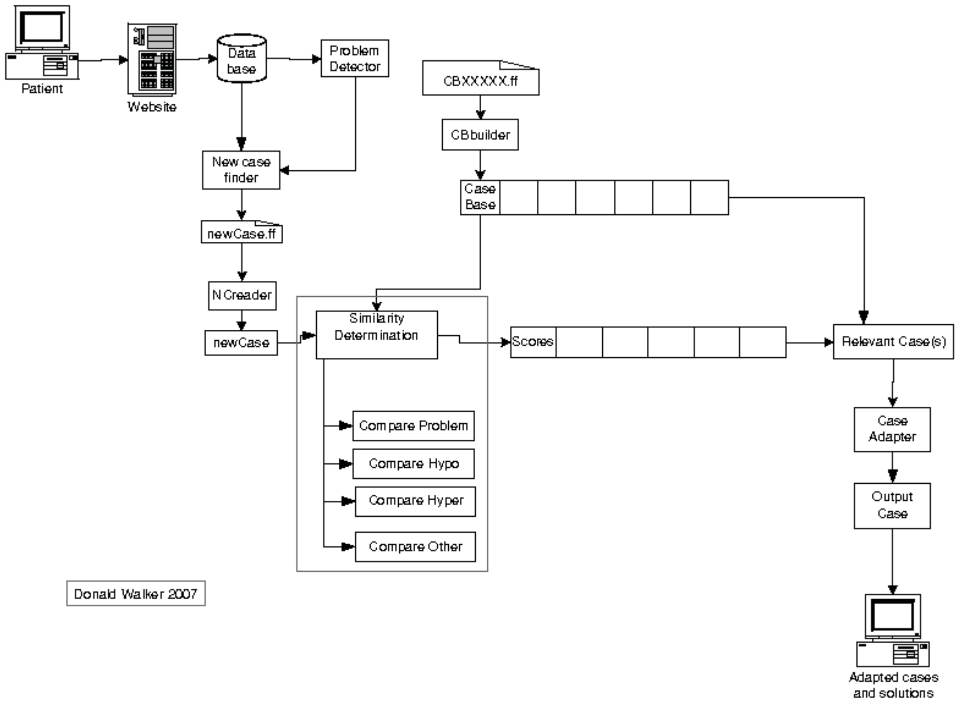

Figure 4.10: Software level overview of IDSDM.

portant in tailoring solutions to the individual experiencing the newly encountered problem. While the solution suggested by the returned case gives the general idea of what needs to be done, such as increasing the carbohydrate ratio or lowering insulin sensitivity, each patient's physiological state is different. The solutions must be adjusted to take into account each patient's high and low blood glucose targets, insulin sensitivity, carbohydrate ratio, lifestyle and other factors.

Another module planned for future development is the data entry automation module. Currently, patients spend somewhere between 30 minutes to one hour per day entering their glucose management data into the patient database (Maimone, 2006). Although much of this data is currently stored in the patients' insulin pumps, there is currently no interface available for automatically extracting the data and entering it into the patient database. Researchers involved with the IDSDM project are working with pump manufacturer Medtronic MiniMed to develop such an interface to be used with this module.

Once all modules are in place and the system is consistently delivering good therapeutic advice for the management of blood glucose levels, IDSDM will become a stand-alone intelligent decision support system. This system will have a wide array of possibilities for implementation. It is feasible for the system to be implemented in the short term future into a portable device worn or carried by patients. In the more distant future it may be possible to integrate the system into an artificial pancreas-like device for automatically managing patients' blood glucose levels much like an artificial pacemaker automatically regulates a patient's heartbeat.

Factors affecting the feasibility of use of IDSDM

Computers have been used in the medical field since the early 1960s; however, their primary function during that period was to perform administrative and fiscal duties (Berner et al., 2005). Since that time, vast improvements have been made in computing technology and the use of computers in the medical field has become increasingly more commonplace, although they still perform primarily administrative and fiscal functions. There have been several movements toward the use of electronic health records (EHRs), but they have not caught on for widespread use for various reasons (Ash and Bates, 2005). Despite this, studies indicate that as people become more comfortable with the use of computers in the health care field the likelihood of EHR adoption increases (Berner et al., 2005; Ash and Bates, 2005; Middleton et al., 2005).

The adoption of electronic health records which contain patient information such as age, weight, blood glucose targets, insulin sensitivity, carbohydrate ratio and other data needed for the operation of IDSDM would be a major advancement for the ease of use of the system. Such a collection of information would facilitate the population of a patient database which could be used in conjunction with the case library to detect and solve many problems in glucose management.

It has been noted that individuals born after 1980 are the first generation to view computers as a household appliance, much like a television or refrigerator (Tapscott, 1998). This generation has been immersed in such technology in the home, school, workplace and many other aspects of everyday life and is more at ease around computers than any other generation. This increased familiarity and comfort with computers has lead to an increasing trend of people turning to the Internet as a source of health information, often as a supplement to regular health care by a physician and occasionally as a replacement for such.

These trends in the acceptance of computers for health care purposes, along with positive patient feedback regarding the IDSDM system (Maimone, 2006) suggest that diabetic patients would view a system such as the finished IDSDM as a valuable tool for assisting with blood glucose management.

Chapter 5

RELATED RESEARCH

This chapter covers related research and technology in both the fields of Case-Based Reasoning and diabetes care. The former will be further divided into general CBR systems and CBR systems used in the health-care system. The latter will consist of a discussion of the state of the art of diabetes care and management technology including insulin pumps and blood glucose monitoring devices as well as research projects aimed at the care and management of diabetes mellitus.

5.1 CBR Research

5.1.1 General CBR Systems

NODAL

NODAL is a CBR system developed at the Trinity and University Colleges in Dublin for the purpose of diagnosing electronic faults.(Cunningham et al., 1998) The researchers involved with this project posit that a problem with using CBR for diagnosis is the dependency upon a complete case description to be present prior to case-retrieval. The authors go on to state that this requirement is often not practical due to the fact that a case may be characterized by a large set of data, not all of which is easily obtainable and possibly not even necessary for retrieval. Thusly, they conclude that the number of features required to determine an accurate diagnosis should be minimized.

To achieve this reduction of required features, the developers of the system propose using "Iterative Case-Based Reasoning" (I-CBR). The idea behind I-CBR is to initially define the case by the easily obtainable, or already known, features and perform a search of the case base as usual. If this initial search yields a unique match, then the I-CBR cycle is ended and the retrieved case is returned as the match. However, if the initial search criteria are too general to result in a unique case match, then more features must be specified to narrow the result set. The authors state that

> "I-CBR works as follows:
>
> - Step 0: Generate a candidate set of cases based on initial features available. The operator provides the system with the values of the free features for the target case. The candidate set is made up of cases that match on these features.
> - Step 1: Select the most discriminating expensive feature in the candidate set.
> - Step 2: Query the operator for the value of that feature in the target case.
> - Step 3: Narrow down the candidate set based on this information. (i.e. Eliminate cases that cannot match this feature, cases for which the value of this feature is unknown are allowed to remain in the candidate set.)
> - Step 4: Repeat from 1 until a unique diagnosis remains."
>
> (Cunningham et al., 1998)

Like the case descriptions in the research of Cunningham et. al., the case descriptions for the IDSDM system contain many parameters; on the order of 150 features may be used. However, the majority of the parameters used for case description are readily available from the patient information database. Also, the number of features directly compared for case retrieval for IDSDM is only on the order of 20 while the rest are used for calculations and output. Thus, a complete comparison of all relevant features is not computationally expensive enough to warrant the use of such an iterative approach. Perhaps once more domain knowledge is gained and the effects of other parameters on glucose management is observed such an approach may become more useful.

CLAVIER

Another example of a non-medical case-based reasoner which is used in industry is CLAVIER, a CBR system designed to achieve the efficient loading of composite material for curing in an autoclave (Hinkle and Toomey, 1995). The reason for the development of this system was to eliminate or reduce the number of low-quality aerospace product components manufactured by the Lockheed company due to uneven curing as a result of poor heat distribution for various components in the oven. This problem arises due to the inherent heating characteristics of each mold based on the size and shape of the mold as well as its position in the oven. These factors lead to the mold and product heating up and cooling off at varying rates, which can lead to reduced quality of the part.

The input to the retrieval mechanism is two-fold. The first input is the case memory of autoclave loads that have been previously run, which consists of the context of the problem (tables used, their positions in the oven, molds used and their positions on the tables) as well as the results of using that configuration. The second input to the retrieval mechanism is the list of parts which need to be manufactured at execution time.

The reasoner specifies three criteria for matching:

> "(1) to maximize the number of needed parts that the load will manufacture, (2) to minimize the number of unmatched (extraneous) parts that the load contains, and (3) to maximize the quality of the load (determined by part compatibility)."(Hinkle and Toomey, 1995)

If an exact match is found in memory, that result is returned; otherwise, several results may be returned ranked by similarity as determined by CLAVIER.

The researchers involved with CLAVIER state that they believe CBR to be a good choice for the autoclave-loading domain due to the fact that they can never be sure what the results will be when curing products using new molds in different configurations. In these situations, they must simply apply what they have learned from using similar configurations in the past. The diabetes domain with which IDSDM deals is similar in that, for each glycemic problem experienced by each diabetic patient, the patient's unique physiology and the number of possibly contributing factors is great enough that a physician cannot be 100% certain of the outcome of a particular treatment. In these situations, the CBR methodology is quite helpful in that it examines these overwhelming contributing factors and considers them in its comparison, making it especially well suited for both domains.

5.1.2 CBR in Medicine

CASEY

CASEY (Koton, 1989) is a system designed by Phyllis Koton at the Massachusetts Institute of Technology which combines case based reasoning with model based reasoning in order to diagnose heart failures. The system is built on top of the Heart Failure Program (Long et al., 1984), which is also the source of the cases in the casebase, and adds case based reasoning functionality. Koton describes CASEY as a system “that uses case-based reasoning to recall and remember problems that it has seen before, and uses a causal model of its domain to justify re-using previous solutions and to solve unfamiliar problems” (Koton, 1989). The memory structure used in CASEY is a discrimination network as described in section 3.1.2.

CASEY begins with patient symptoms as input to the system and uses this input to create a causal network of internal states that may have lead to the current symptoms. Koton cites several major advantages of CASEY. The first advantage is the fact that the system first identifies which features are important to matching by using the causal model. Another advantage is that the system can prove the applicability of a suggested solution by analyzing differences between the retrieved case and the new case based on the causal model. Koton also cites the speed advantage of CASEY over model-based-only systems since CASEY remembers previous similar situations and modifies the solutions that were used in those cases. The other major advantage of CASEY that Koton lists is its ability to recognize when it has not seen any cases similar enough to the input to suggest a solution. In this case the system would simply pass the problem on to the existing model-based reasoning functionality of the Heart Failure Program.

Testing of the CASEY system has shown it to give results comparable to the model-based expert system for the domain but with an increase in efficiency of several orders of magnitude (Koton, 1989). Koton also posits that the techniques developed for the system are domain independent and can be used in other domains having similar form.

Unfortunately, at this time not enough is known about the domain of diabetes care to create a causal model of its associated symptoms and problems and, as was mentioned in Chapter 3, no good discrimination network could be created based on the data collected during the IDSDM clinical study. The IDSDM system does however make use of some of the other advantages of CASEY such as recognizing when it has not seen any cases similar enough to recommend a solution.

ICONS

ICONS (Schmidt et al., 1999; Heindl et al., 1997) is a CBR system designed to prescribe antibiotics to patients in intensive care who have bacterial infections. The prescribed antibiotic regimen should satisfy the medical and economic constraints entered into the system. The memory organization in ICONS is represented in a hierarchical cluster of cases as described in section 3.1.2. The ICONS system is similar to IDSDM in that it seeks to give advice for a specific, acute medical problem in the form of corrective action.

ICONS was created in order to overcome the problem of intensive care patients being prescribed antibiotics with potentially harmful contraindications. Conditions in intensive care units are usually fast paced and may be lacking in complete patient data and as such it is necessary to prescribe patients with bacterial infections antibiotics that will have the greatest benefit with the least detriment and to do so in the least amount of time. This is the goal of ICONS.

The ICONS process is described as "Find all possible solutions and reduce them using the patient's contraindications and the complete coverage of the calculated pathogen spectrum" (Schmidt et al., 1999). More specifically, a list of antibiotics to which the patient's bacterial infection is susceptible is created. A subset of this list is then found by applying the constraints of the patient's contraindications and the desired sphere of activity of the medication. From this subset is chosen a regimen of antibiotics which will cover the entire expected bacterial spectrum.

ICONS seeks to speed up this process with the use of CBR. ICONS, like IDSDM, uses a two step retrieval process. In the first step a portion of the hierarchically organized case base is chosen. The portion chosen must consist of cases sharing two factors with the newly input case: the patient group and the infected organ system. Similarity functions are then applied to these retrieved cases based on contraindications. Highly matching cases are then adapted when necessary and an antibiotic regimen is suggested.

PROTOS

PROTOS (Bareiss, 1988; Porter et al., 1990) is a classification system used for diagnosing hearing disorders. This system is a case-based approach to concept learning for heuristic classification (Porter et al., 1990). PROTOS, whose case base is filled with domain knowledge, learns concepts by retaining exemplars and classifies new cases by matching them to the exemplars. The authors cite two main obstacles to this approach: how to measure similarity between the new case and exemplars and how to efficiently find a matching exemplar. These problems are overcome by augmenting the exemplar cases with domain knowledge and indexing data. The authors state that their approach allows the system to begin operation completely incompetent and progress to expert competence by learning domain specific concepts.

PROTOS determines proper classification by determining explanations. Explanations are created by making "chains" of relationships, which are pieces of domain-specific knowledge which tie together the features and categories of the cases. Explanations are accepted once enough relationships have been formed to bring the strength of the explanation above a threshold value.

The system was trained with 200 cases by an expert audiologist and has used those cases to learn enough domain knowledge to be able to make hearing disorder classifications comparable to an expert human audiologist.

5.2 Diabetes Research and Technology

5.2.1 Diabetes Systems

Telematic Management of Insulin Dependent Diabetes Mellitus (T-IDDM)

While not initially making use of case based reasoning, the T-IDDM system (Bellazzi et al., 2002), funded by the European Union (EU), is a telemedicine system designed for decision support for insulin dependent diabetic patients. The system consists of two modules. The patient unit (PU) is used by the patient to transmit data from their blood glucose monitoring device to the hospital's database via Internet or telephone. The medical unit (MU), used by physicians, provides tools for data visualization, data analysis, decision support as well as allowing the physician to transmit messages and therapeutic advice to the patient (Bellazzi et al., 2002).

The decision support module of T-IDDM, which implements rule-based reasoning (RBR), consists of four tasks, each mapped to a specific set of rules. The tasks are: *data analysis* which indicates the patient's response to the current insulin therapy regimen; *problem identification* fires based on the results of the data analysis and indicates either a hypoglycemic or hyperglycemic episode throughout the day; *sug-*

gestion selection proposes a set of solutions and chooses one based on the problem, time of problem and the pharmokinetics of the insulin; and *therapy revision*, in which the RBR system adjusts the current insulin therapy in accordance with the suggested solution (Bellazzi et al., 2002).

Evaluation of the initial T-IDDM system showed (Bellazzi et al., 2002) that it sometimes suggested solutions that were not well suited for the exact situation being experienced at the time. As such, preliminary testing was performed on a upgraded version of T-IDDM which made use of CBR in these situations. This version presents past similar cases to the user and uses them to alter rule behavior and parameters to tailor the suggested solution to better fit the new problem. Evaluation also showed promising results (slightly lower blood glucose levels and HBA1C levels), although the system's creators acknowledged that it "is essentially a feasibility study, due to the small number of patients involved, and to the absence of a control group" (Bellazzi et al., 2002).

There are several similarities and dissimilarities between T-IDDM and IDSDM. Both systems seek to provide decision support for insulin dependent diabetic patients, primarily those with Type 1 diabetes. Both systems also base their decision support in part on readings from a patient's self blood glucose monitoring. Beyond these similarities are many significant differences.

One major difference is the type of data used for decision support and the frequency of data analysis. The data used for decision support in T-IDDM consists of values from three or four blood glucose readings per day, which are transmitted from the PU to the MU every seven to ten days. The data used for decision support in IDSDM is based on many more features, which may be more frequently analyzed by the problem detection software.

Another main difference is the level of involvement required of both the physician and the patient in use of the system. T-IDDM requires that the patient periodically transmit data to the MU and that the physician manually review the data and therapy adjustments suggested by the system. IDSDM requires that the patient enter six to ten blood glucose readings per day, plus extensive daily lifestyle data. It also incorporates data from continuous blood glucose monitoring systems, which provide 288 data points per day. IDSDM simply gives the physician the suggested solution from the most highly matching case, at present, although patient specific therapy adjustments will be recommended in the future.

The third major difference between the two systems is the method of reasoning used for decision support. The decision support methodology of IDSDM consists solely of case-based reasoning whereas T-IDDM implements multi-modal reasoning, using primarily rule-based and model-based reasoning and, in preliminary test versions, using case-based reasoning as a supplement.

Computer Assisted Meal Related Insulin Therapy (CAMIT)

The CAMIT study (Schrezenmeir et al., 2002) did not involve the use of any artificial intelligence methods. CAMIT's goal was to provide suggested adjustments to insulin dosages for diabetic patients via a pocket computer that was carried by the patient. During the study, 25 subjects carried the pocket computer and used the recommended adjustments, while a control group of another 25 subjects continued to use their usual regimen of multiple subcutaneous injections (MSI) of insulin with set prescribed dosages. The insulin dose calculations for those using the CAMIT pocket computer are based on multiple factors including the patient's current blood glucose level, insulin sensitivity, carbohydrate ratio, the expected carbohydrate intake of the upcoming meal and other factors related to the current metabolic state of the patient (Schrezenmeir et al., 2002).

The CAMIT study shows that computer aided insulin management can be effective and that such a system is feasible. It also demonstrates that patients are open to the idea of using suggestions for their diabetes management from calculations made by such systems.

5.2.2 Commercially Available Diabetes Care Technology

There is a great deal of technology used in the field of diabetes care, although little of it makes use of artificial intelligence as does the IDSDM system. For the past 20 years, Medtronic MiniMed has sold many products which have helped hundreds of thousands of patients manage their diabetes, making them the world's largest vendor in this area (Medtronic MiniMed, 2007f).

The first Medtronic MiniMed product of interest is their continuous glucose monitoring system (CGMS), which was used in the IDSDM study. This unit provides a blood glucose reading every five minutes, up to 864 readings in the 72 hour period for which it is worn by the patient (Medtronic MiniMed, 2007c). These frequent readings can show patterns which might be missed by the infrequent fingerstick measurements normally performed by the patient. For instance, a patient could easily be within his or her suggested blood glucose range when the fingerstick measurements are taken (generally before each meal and before going to bed) but could be having wide swings in glucose levels between readings and never know about it. This system allows the user to "compile and analyze comprehensive glycemic-trend data in a logical, graphical format - increasing the depth of information on which you can base clinical decisions" which can greatly improve the patient's overall diabetes management (Medtronic MiniMed, 2007c).

The next offering of note from Medtronic MiniMed is the insulin pump. All patients in the IDSDM study wore Medtronic Minimed insulin pumps. Medtronic MiniMed offers several different models of insulin pumps (Medtronic MiniMed, 2007f). Insulin pumps are small devices worn like a pager. The unit contains a small vial of insulin that can be delivered throughout the day in the form of basal or bolus doses of insulin depending on the parameters entered such as the patient's insulin sensitivity, carbohydrate ratio and number of grams of carbohydrate consumed during a meal. The units also have the option of setting a temporary basal rate to deliver a smaller amount of continuous insulin or to completely suspend insulin delivery during periods of heavy activity or hypoglycemia. The insulin is delivered to the patient via one of several options for infusion sets which may be tailored to the individual patient for maximum comfort and effectiveness (Medtronic MiniMed, 2007d).

These systems do not contain any artificial intelligence, but make use of mathematical formulas and graphical data displays for decision support. Many of MiniMed's insulin pumps come with the Bolus Wizard Calculator (Medtronic MiniMed, 2007a), which automatically calculates the bolus dose of insulin based on current blood glucose levels, insulin sensitivity, carbohydrate ratio and the anticipated carbohydrate content of a meal. Pumps equipped with the Bolus Wizard Calculator can help patients better manage their glucose levels by reducing math errors and minimizing the number of correction boluses required throughout the day (Medtronic MiniMed, 2007a). The pumps also offer customization of up to eight different settings per day for blood glucose targets, carbohydrate ratios and insulin sensitivities (Medtronic MiniMed, 2007a) as well as multiple bolus dose delivery options including normal bolus, square wave bolus and dual wave bolus (Medtronic MiniMed, 2007e).

The newest offering from MiniMed is the Paradigm system, which integrates real time continuous glucose monitoring with an insulin pump (Medtronic MiniMed, 2007e). Features of this combination include the ability to display updated glucose values every five minutes, display three- or 24-hour glucose trend graphs, indicate in which direction and how fast blood glucose levels are moving, and alert patients to hypoglycemic and hyperglycemic episodes with an audible or vibrating alarm (Medtronic MiniMed, 2007e). This system is advertised as being ideal for those who are prone to nighttime lows, or post-meal highs, morning highs or for those who are pregnant, or planning to get pregnant (Medtronic MiniMed, 2007e). The system also allows the user to plot historical glucose level trends using the online CareLink software (Medtronic MiniMed, 2007b).

Chapter 6

SUMMARY AND CONCLUSION

This thesis has presented a metric for similarity determination and case retrieval for an intelligent decision support system intended to help diabetic patients manage their blood glucose levels. The IDSDM system that is a result of this and other research will help these patients better regulate their blood glucose levels through the use of an artificial intelligence approach called case-based reasoning. Case-based reasoning uses the solutions from problems encountered and solved in the past to solve newly encountered problems.

The research used an Ohio University IRB approved study which collected and analyzed glucose management data from 20 diabetic patients in order to populate an initial case library. This interdisciplinary research is part of a collaborative effort between Ohio University's School of Electrical Engineering and Computer Science in the Russ College of Engineering and Technology and the Appalachian Rural Health Institute Diabetes/Endocrine Center of the College of Osteopathic Medicine.

Contributions of this thesis include:

- designing and implementing logical structures to contain case descriptions in collaboration with fellow graduate student Eric Flowers
- designing and implementing the metric for similarity determination and case retrieval
- extensive testing of the similarity metric's performance.

Case descriptions were designed and implemented using a structure of hierarchical Java classes. Comparison functions were created to determine the similarity between features as well as the importance of such similarities to a match. The metric was evaluated by three physicians specializing in the treatment of diabetes.

Evaluations of the performance of the IDSDM system's similarity determination and case retrieval modules indicate that the system returns very similar or somewhat similar cases along with useful advice about 80% of the time. Several suggestions were given for future work which may improve similarity determination and case retrieval in IDSDM. These suggestions include increasing the size of the case library, adding more problem categories or subtypes, and comparing additional features during similarity determination.

This thesis also presented related work in both the fields of case based reasoning and diabetes care. The majority of technology used in the field of diabetes care involves insulin delivery and blood analysis and makes use of mathematical formulas rather than intelligent reasoning processes. Positive patient feedback regarding the IDSDM system as well as trends in consumer acceptance of computers for use in the health care field suggest that this work could lead to a practical tool for patients with diabetes as well as increase knowledge in the fields of artificial intelligence and medicine.

Bibliography

Aamodt, A. and Plaza, E. (1994). Case-based reasoning: Foundational issues, methodological variations, and system approaches. *Artificial Intelligence Communications*, 7(1):39–59.

American Diabetes Association, editor (2002). *American Diabetes Association Complete Guide To Diabetes*. Bantam Books, New York, 3rd edition.

Ash, J. S. and Bates, D. W. (2005). Factors and forces affecting EHR system adoption: Report of a 2004 ACMI discussion. *Journal of the American Medical Informatics Association*, 12(1):8–12.

Bareiss, E. R. J. (1988). *Protos: a unified approach to concept representation, classification, and learning*. PhD thesis, University of Texas.

Bellazzi, R., Larizza, C., Montani, S., Riva, A., Stefanelli, M., d'Annunzio, G., Lorini, R., Gomez, E. J., Hernando, E., and Brugues, E. (2002). A telemedicine support for diabetes management: the T-IDDM project. *Computer Methods and Programs in Biomedicine*, 69(2):147–161.

Berner, E. S., Detmer, D. E., and Simborg, D. (2005). Will the wave finally break? a brief view of the adoption of electronic medical records in the united states. *Journal of the American Medical Informatics Association*, 12(1):3–7.

Bolli, G. B., Perriello, G., Fanelli, C. G., and DeFeo, P. (1993). Nocturnal blood glucose control in type 1 diabetes mellitus. *Diabetes Care*, 16(suppl3):371–389.

Cheeseman, P., Kelly, J., Self, M., Stutz, J., Taylor, W., and Freeman, D. (1988). Autoclass: A Bayesian classification system. In *Proceedings of the Fifth International Machine Learning Workshop*, New York. Morgan Kaufmann.

Cunningham, P., Smyth, B., and Bonzano, A. (1998). An incremental retrieval mechanism for casebased electronic fault diagnosis. *Knowledge Based Systems*, 1(3-4):239–248.

Domeshek, E. A. (1992). *Do the Right Thing: A Component Theory for Indexing Stories as Social Advice*. PhD thesis, Yale University.

Fisher, D. H. (1987). Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, 2:139–172.

Heindl, B., Schmidt, R., Schmidt, G., Haller, M., Pfaller, P., Gierl, L., and Pollwein, B. (1997). A case-based consiliarius for therapy recommendation (ICONS): computer-based advice for calculated antibiotic therapy in intensive care medicine. *Computer Methods and Programs in Biomedicine*, 52(2):117–127.

Hinkle, D. and Toomey, C. (1995). Applying case-based reasoning to manufacturing. *AI Magazine*, 16(1):65–73.

Jain, A. (2002). A case-based tool for treatment of behavioral problems associated with Alzheimer's disease. Master's thesis, Ohio University.

Kolodner, J. (1993). *Case-Based Reasoning*. Morgan Kaufmann, San Francisco.

Koton, P. (1989). *Using Experience in Learning and Problem Solving*. PhD thesis, Massachusetts Institute of Technology.

Leake, D. B. (1992). *Evaluating Explanations: A Content Theory*. Lawrence Erlbaum, Mahwah, NJ.

Long, W., Naimi, S., Criscitiello, M., Pauker, S., and Szolovits, P. (1984). An aid to physiological reasoning in the management of cardiovascular disease. In *Computers in Cardiology*.

Lopez de Mantaras, R., McSherry, D., Bridge, D., Leake, D., Smyth, B., Craw, S., Faltings, B., Maher, M. L., Cox, M. T., Forbus, K., Keane, M., Aamodt, A., and Watson, I. (2006). Retrieval, reuse, revision and retention in case-based reasoning. *The Knowledge Engineering Review*, 20(3):215–140.

Maimone, A. (2006). Data and knowledge acquisition in case-based reasoning for diabetes management. Master's thesis, Ohio University.

Marling, C., Shubrook, J., and Schwartz, F. (2007a). Towards case-based reasoning for diabetes management. Workshop Proceedings of the International Conference on Case-Based Reasoning, Belfast, Northern Ireland.

Marling, C., Shubrook, J., Schwartz, F., Walker, D., Flowers, E., and Jones, T. (2007b). Intelligent decision support system for diabetes management. Accepted for the Seventh Annual Diabetes Technology Meeting in San Francisco, California.

Marling, C. R., Shubrook, J. H., Miller, W. A., Maimone, A. J., and Schwartz, F. L. (2007c). Intelligent decision support for type 1 diabetics on insulin pump therapy. In *67th Scientific Abstracts Book*. The American Diabetes Association.

McCarthy, J. (2000). What is artificial intelligence? http://www-formal.stanford.edu/jmc/whatisai/.

Medtronic MiniMed (2007a). Bolus wizard calculator. http://www.minimed.com/products/insulinpumps/features/boluswizard.html.

Medtronic MiniMed (2007b). CareLink therapy management software. http://www.minimed.com/products/software/index.html.

Medtronic MiniMed (2007c). CGMS system gold: Continuous glucose monitoring system overview. http://www.minimed.com/products/cgms/index.html.

Medtronic MiniMed (2007d). Infusion sets. http://www.minimed.com/products/infusionsets/index.

Medtronic MiniMed (2007e). Insulin pumps. http://www.minimed.com/products/insulinpumps/ind

Medtronic MiniMed (2007f). Product information. http://www.minimed.com/products/index.html.

Michalski, R. S. and Stepp, R. (1983). Learning from observation: Conceptual clustering. *Machine Learning*, 2:331–363.

Middleton, B., Hammond, E., Brennan, P., and Cooper, G. F. (2005). Accelerating U.S. EHR adoption: How to get there from here. recommendations based on the 2004 acmi retreat. *Journal of the American Medical Informatics Association*, 12(1):13–19.

Porter, B., Bareiss, R., and Holte, R. (1990). Concept learning and heuristic classification in weak-theory domains. *Artificial Intelligence*, 45(1-2):229–263.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1(1):81–106.

Quinlan, J. R. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann, San Francisco.

Quinlan, J. R. (1996). C4.5 software. http://www.rulequest.com/Personal/.

Rich, E. and Knight, K. (1991). *Artificial Intelligence*. McGraw-Hill, New York, 2nd edition.

Russell, S. and Norvig, P. (2003). *Artificial Intelligence A Modern Approach*. Prentice-Hall, Upper Saddle River, NJ, 2nd edition.

Schank, R. and Osgood, R. (1990). A content theory of memory indexing. Technical Report 2, Northwestern University, Institute of Learning Sciences.

Schmidt, R., Pollwein, B., and Gierl, L. (1999). Case-based reasoning for antibiotics therapy advice. In *ICCBR '99: Proceedings of the Third International Conference on Case-Based Reasoning and Development*, pages 550–559, Berlin. Springer-Verlag.

Schrezenmeir, J., Dirting, K., and Papazov, P. (2002). Controlled multicenter study on the effect of computer assistance in intensive insulin therapy of type 1 diabetics. *Computer Methods and Programs in Biomedicine*, 69(2):97–114.

Smyth, B. and Keane, M. (1995). Remembering to forget: A competence-preserving deletion policy for case-based reasoning. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, Montreal, Canada.

Stephenson, J. and Schernthaner, G. (1989). Dawn phenomenon and Somogyi effect in IDDM. *Diabetes Care*, 12(4):245–251.

Stratton, I. M., Adler, A. I., Neil, A. W., Matthews, D. R., Manley, S. E., Cull, C. A., Hadden, D., Turner, R. C., and Holman, R. R. (2000). Association of glycaemia with macrovascular and microvascular complications of type 2 diabetes (UKPDS 35): prospective observational study. *Brittish Medical Journal*, 321:405–412.

Tapscott, D. (1998). *Growing Up Digital: The Rise of the Net Generation*. McGraw Hill, New York.

The Diabetes Control and Complications Trial Research Group (1993). The effect of intensive treatment of diabetes on the development and progression of long-term complications in insulin-dependent diabetes mellitus. *The New England Journal of Medicine*, 399(14):977–986.

Ybarra, M. L. and Suman, M. (2006). Help seeking behavior and the internet: A national survey. *International Journal of Medical Informatics*, 75(1):29–41.