GIDS for Chronic Disease Visualization: Implementation of Chronological Clustering as Workflow System

Fawaz AL-Hazemi

Electrical Engineering Department, KAIST
291 Daehak-ro, Yuseong-gu, Daejeon 305-701, South Korea (Building# E3-2, Room# 6-3211)
fawazhazemi@ieee.org; fawaz@kaist.ac.kr

Abstract- **Chronic disease is linked to patient's' lifestyle. Therefore, doctor has to monitor his/her patient over time. This may involve reviewing many reports taken over short and long periods. Computer applications made it possible to visualize these reports on single displayer such as a timeline-based visualization tool. However, there is a limitation of studying the diabetes patient's history to find out what was the cause of the current development in patient's condition. In this paper, we propose a workflow system which uses the Grid-based Interactive Diabetes System (GIDS) to support diabetes analysis. Technically, the workflow uses an agglomerative clustering algorithm as clustering correlation algorithm which reduces the revision of long reports and focus on vital incidences. However, if doctor demands to review the full reports, the workflow displays the original reports. Through basic evaluation experiment, we were able to demonstrate the usability of the system as chronic patient's multi visualizer.**

Keywords- Bioinformatics; Chronological Clustering; Grid Computing; Diabetes; Problem Solving Environment; Timeline Visualization

I. INTRODUCTION

Computer aided tools such as Computer Aided Detection/Diagnosis (called CADe or CADx) differ according to the disease itself and patient's conditions. For example, diabetes requires continuous monitoring on patient's blood sugar. Self-check tools such as ACCU-CHEK used by patients in a daily basis are capable of uploading recorded data to medical repository online. Moreover, diabetes may need the observation of other conditions such as heart, fat, cholesterol and Magnetic Resonance Imaging (MRI). Each one of these conditions needs an appropriate application tool. Combining these tools has led us to what is called nowadays e-healthcare. After screening the outcome from these application tools, a medical doctor would precisely identify the cause of the current patient's condition. Although a medical doctor has the knowledge and the necessary analysis processes for any type of diseases, it is complex task to a human to combine and evaluate all application tools outcome. In addition, the historical analysis of a chronic disease patient is essential and without the assistance from computer machine it is difficult. Therefore, the crucial problem in e-healthcare systems is not only the integration of application tools but the intelligent visualization for these outcomes.

In literature, there are many proposed approaches undertook chronic diseases visualization as a main focus of the e-healthcare system. One approach is patient data visualization; which is integrating patient's history into single viewer as in [1-3]. For example, Bui et al. [1] tried to visualize patient's history in a problem centric optimizer, where a TimeLine based visualization tool has been developed for that purpose. Using the TimeLine tool, medical doctor is able to classify the causes and modifying the treatment. However, the TimeLine approach follows two steps to accomplish; basically the data federation and organizing the federated data. These two steps are accomplished according to their temporal acquisition. After completing these two steps, a customized display is generated to the medical doctor (or user). Combi et al [2] suggested temporal abstractions using the visual language. Patient's historical data are organized according to the consistency of his/her condition. After this organization, doctors are able to visually query patient history using basic Boolean operations (AND, OR, and NOT). The limitation of this suggestion is that doctors need to add efforts to build a query to extract useful data. A leak of constructing good queries (by doctor) will result in poor feedback by the system. Tague et al. [3] proposed an interactive visualization for vital signs. Patient's historical data are presented as trends over time where vital signs are located, and they are grouped according predefined periods. System's user can optimize the visualization by zooming in or zooming out through controlling the time span (called lens). The proposed work helps doctor to visualize the vital signs. However, minor signs are disappeared in the visual trends which require doctor to zoom in repeatedly till these minors are displayed.

On the other hand, there is an approach that uses a content-based healthcare record to help simplify the diagnosis [4-8]. Toyoda et al. [4, 5] proposed tracking entry system called Sakura-view, which was an order-entry technique to keep tracking the order-entry. All medical data are oriented based on order and order-related data about patients. This approach is generally used in medical to diagnosis patients by referring to their historical data. However, this type of systems had limitations in visualizing (on one hand) vast historical data and (on other hand) low frequent data might be hidden by high frequent data. Consequently, the content-based approach forces medical doctors to put extra efforts into understanding the vast data.

A system for remote patient monitoring has been proposed [9, 10]. Moreover, a personal or self-monitoring system for

DOI: 10.5963/LSMR0402001

diabetes patients is available in literature [11]. These systems are promising in healthcare industry but they are missing the careful computation of patient data. For instance, in our work, we used the Gower's similarity method as in [12], which has the advantage of truncating the computations of missing portion of the data without affecting the accuracy of the calculations. In addition, different weighing algorithms to cluster the data, such as the algorithms mentioned in the methodology section (section 3), will definitely have different clustering.

A diabetes prediction modulation has been proposed to help assist patients and doctors in controlling diabetes condition. For example, authors in [13-15] have designed neural network models (NNM) for real-time prediction of patient condition. In addition, authors in [16] have suggested the use of Autoregressive (AR) modulation. The benefit of having such modulation of patient's condition will add intelligent guidance to patient's treatments. Nevertheless, these contributions focused on providing self-adjustment of patient's treatments but not a way of visualizing patient's historical data.

To our knowledge, none of the previously mentioned approaches showed concerns over backend computation developments, regardless of the flexible computing environment. Diverse historical data about a diabetes patient should be processed all at once by different application tools and in the same workspace. In addition, similar works are proposed to support medical decisions as in [6-8]. GIDS has the capability of accommodating application tools as services (applications ready installed in grid computing repository) and they are on hand to users via the workflow compositor [17-19]. These capabilities enable the user to have variety of application tools to assist in the decisions.

Apparently, a worth to mention that there are several research solutions which focused on the involvement of the backend computation developments to improve the services such as in [20, 21]. MammoGrid [20] is one of the Grid platform solutions supporting radiologists to diagnosis cancers with mammography. MammoGrid was based on Service Oriented Architecture (SOA) and used the web service communication (SOAP) [22]. Nevertheless, MammoGrid is not focused to time-based visualization. And [21] is a Grid based platform that utilized the real-time image processing of MRI intra-operatively. The focus of this research is to achieve a very short time of image processing and feedback to doctors in the operation surgery, therefore a load balancing and fault tolerance were considered in the development. The outcome of this system is useful during the operation surgery and it is not related to visualizing patient data. In our work, we tried to adopt the innovation of having high performance computing (HPC) such as grid computing, and improve the integration of data with the service oriented architecture [22]. It is vital when there are system users (doctors) who has limited computation capabilities device such as smart device. Our proposal could off load the heavy computation from smart devices to HPC backend infrastructure.

To formalize this paper, the following subsections are our motivations, contributions and the structure of this paper.

A. Motivations

The health care system for chronic diseases' patients such as diabetic patients has divers' potentials areas for developments. To name some, any chronic disease has various exams and laboratory tests to analysis, bioinformatics applications to use, collaboration between different clinics and clinicians to consider, and timeline displayer to review patient history. These areas are wide and they could be developed according to different targeted goals. This would yield in different implementations of the same developed area. The vast of data generated from personal examinations (e.g. blood sugar and blood pursuer) and clinical examinations (e.g. laboratory tests and cardiology examinations) must be stored, retrieved and analysed. In a chronic disease healthcare system, there is high expectation of computation bottleneck when there are huge data. Besides, these generated data could be collected based on different periodicals. For example, the blood sugar is monitored daily while HbA1c is examined (on an average) every 3 months. Therefore a multi variation analysis is required.

The complexity will be escalated when medical doctor requires different views of patient's trends to track any incremental development. It is hard to fit all collected data in one single displayer. As a result, a correlation and clustering are required for best-fitting the data into single viewer. In the case of remote consultation, doctors who are outside office may use a smaller displayer such as the one available on smart phones and tablets, but they will experience computation and visualization limitations.

B. Paper Contributions

Our proposed solution is a Grid-based workflow system that supports visualizing diabetes patient's history in an interactive mode. The workflow is supported by backend Grid-based Interactive Diabetes System (GIDS) [23, 24]. The system provides a problem solving environment (computational environment) to study diabetes patient (and could be extended to any chronic disease patients). It collects, processes and displays patient's data as trends into single displayer. GIDS provide varieties of optimized timeline views for patient's condition. These variations are controlled by an agglomerative clustering algorithm [25-27]. Ultimately, doctors are able to view results of the data analysis into different scales and can promptly monitor their patients.

Apparently, in this paper, we do not provide how to correlate the relationships between different data type, and we assumed them to be available as numerical numbers. For example, results from urines tests, blood sugar tests, Electrocardiograph (ECG) tests have different data types, and they must be compared to their references prior engaged them in the analysis. Therefore, we assumed that such data conversion must be done prior our system proceed. Currently, we are defining a task for data

DOI: 10.5963/LSMR0402001

conversion before GIDS workflow, however, it is under development and not yet tested.

C. Paper Structure

The structure of the paper is as follow. In section 2, a background is briefing the diabetic patient's data. In section 3, the methodology of our approach is discussed, where there are two sub sections devoted to similarity calculation methodology and clustering algorithms. Sections 4 and 5 are briefing the GIDS and the structure of the workflow structure used in our system. Section 6 is the basic evaluation we have performed and finally section 7 is concluding this paper.

II. BACKGROUND

Diabetes patient is chronic patient whose daily life style is monitored. Several daily activities must be examined and recorded. In addition, incidents such as blurry vision and dizziness must be recorded once they occur. On the other hand, some examinations are taken periodically at clinic, such as laboratory tests for Urine, HbA1c and cholesterol [28]. Table 1 lists some of these examinations and their descriptions.

TABLE 1 DIABETES EXAMINATIONS

Examination	Definition	Type	Unit	Levels
Urine	Look for glucose and ketones from breakdown of fat	Lab test	N/A	N/A
HbA1c	Monitor how well patients are controlling Blood glucose by checking the Haemoglobin A1c in the cell Check every 3 months	Lab test	%	Normal: Less 5.7% Pre-diabetes: 5.7%-6.4% Diabetes: 6.5% or higher
Fasting blood glucose level	Check the blood sugar when person is fasting (not eating)	Personal test	mg/dL	Normal: 100 -126 mg/dL Diabetes: 126 mg/dL or higher
Oral glucose tolerance test	Check the blood sugar after 2 hours from eating	Personal test	mg/dL	Diabetes: 200 mg/dL or higher
Random (non-fasting) blood glucose level	Random time and accompanied by classic diabetes symptoms (increased thirst, urination, and fatigue). Must be confirmed with a fasting blood glucose level test	Personal test	mg/dL	Diabetes: 200 mg/dL or higher
Blood pressure	Check blood pressure	Personal test	mmHg	Diabetes: 130/80 mmHg or higher
HDL cholesterol	Check High Density lipoprotein (HDL) cholesterol	Lab test	mg/dL	Diabetes: Less than 40 mg/dL
LDL cholesterol	Check Low Density lipoprotein (LDL) cholesterol	Lab test	mg/dL	Diabetes: more than 100mg/dL
Total cholesterol	Check cholesterol in blood	Lab test	mg/dL	Diabetes: more than 100mg/dL
Triglycerides	Check Triglycerides in blood	Lab test	mg/dL	Diabetes: more than 150 mg/dL
Waist circumference	Measure the length of around the waist	Personal test	Inches	Diabetes Men: 40 inches or more Diabetes Women: 35 inches

Patient's history is a set of data instances collected from different media. Each instance has many data entities such as blood sugar, HbA1c and urine tests. For example, personal blood test such as blood sugar and blood pressure could be recorded by patient in a daily base. Moreover, clinical tests such as urine tests, ECG signals and HbA1c tests are taken by specialists in different periods (monthly or every three months). The potential challenges in patient's history are mainly two challenges. First, each test has different data type (see Table 1), that made it hard to compare records in different tests without conversion (raw data). Second, there are many instances that have missing data entities which are caused by the variation in tests due.

III. METHODOLOGY

Putting patient's data into single view using data clustering is meaningless as reported by authors in [29-31]. Therefore, a numerical analysis must be applied prior the clustering (or grouping). Patient's data must be measured in terms of resemblance, and then grouped using temporal and agglomeration procedures. The ordination of resemblance patient's data should be considered. Bui et al. [1] claim the use of chronological clustering but they have used the k-mean algorithm which is not agglomerative. The K-mean is a partitioning algorithm that takes a set of data as one cluster/group. Then it divides that set into subsets according to the similarity (or distance) between entities in the original set. The outcome subsets are repartitioned using k-means to generate more subsets while there is no option or procedure to reorganize the subsets into temporal sequences. Therefore, there must be a time-based algorithm that assists in clustering patient's data chronologically.

As we have reviewed in the previous background section, there are two main challenges in patient's historical data namely multiple data types and missing data entities in data instances. The problem of multiple data types could be solved by reformatting and converting data into more standardized forms. For missing data entities, it is possible to find the similarity between two instances using adequate similarity calculations. In this paper, we use the data without conversion to find the similarity metrics between data instances even if there are missing data entities. To accomplish this procedure, we have to define the methodology of finding the similarity between data instances and explain the clustering algorithm used.

A. Similarity Calculation Method

Any grouping algorithm must define or specify the similarity methodology used among data. For instance, to decide

DOI: 10.5963/LSMR0402001

whether two entities are similar (or having almost similar content), a similarity/distance methodology must be applied to evaluate the similarity ratio. In Legendre et al. [25], there are many similarity methodologies with their similarities coefficients. Similarity calculation between two objects is sometimes evaluated as distance ratio. The distance ratio is calculated using the Euclidean distance or any other mathematical calculations. Then, that distance ratio is subtracted from 1 to yield the similarity ratio as in Eq. (1).

$$S = 1 - D_{norm} \tag{1}$$

Where S is the similarity ratio between 0 and 1, and D_{norm} is the normalized distance ratio between 0 and 1. The most common distance calculation methods is the Euclidean distance that is a Pythagoras method as in the below Eq. (2).

$$D(\mathrm{x}, \mathrm{y}) = \sqrt{\sum_{i=1}^{n} (\mathrm{x}_i - \mathrm{y}_i)^2} \tag{2}$$

Where D is the Euclidean distance between object x and object y, and each object has n entities. Apparently, the Euclidean distance method could possibly lead to inaccurate evaluation of the distance (or similarity) ratio among patient's historical data instances. The following example will illustrate that inaccuracy.

Example 1**:** Let us assume there are three data instances among patient's history namely T_1, T_2 and T_3. At each instance, the patient was recording the blood sugar measurements for both "fasting glucose" and "2 hour glucose". At instances T_1 and T_3, patients have visited the clinic's laboratory to evaluate the percentage of HbA1c in the blood – particularly in the red blood cells. The medical records are listed in Table 2. The question is if we would like to group the intermediate instance T_2 to one of the remaining two instances (T_1 and T_3), which instance must T_2 be grouped with? Practically, instance T_2 is closer to T_3 than to T_1.

First, we have to evaluate the similarities (or the distances) ratios. According to the Euclidean distance Eq. (2), the distance ratios among T_1, T_2 and T_3 are listed in Table 3. And from there, we could theoretically say that the data instance T_2 is closer to T_1 which has lower distance (0.359) than T_3 which has higher distance (0.445). Although the HbA1c was not reported in T_2 instance, T_2 has blood sugar measurements closer to those reported in T_3 instance than to those in T_1 instance. Mathematically, the differences between instances T_2 and T_3 are about 2 and 1 mg/dL in fasting glucose and 2-hour glucose respectively. And if we compare them to the differences between instances T_2 and T_1 which were 3 and 4 mg/dL in fasting glucose and 2-hour glucose respectively, then it is clear that T_2 is closer to T_3.

TABLE 2 EXAMPLE OF THREE DATA INSTANCE (RECORDS) TAKEN FOR DIABETES PATIENT

Data instance	Blood Sugar "fasting glucose" (mg/dL)	Blood Sugar "2 hour glucose" (mg/dL)	HbA1c (%)
Ref.	**90**	**125**	**4.5**
T_1	117	137	6.1
T_2	120	141	0 (null)
T_3	122	142	6.5

TABLE 3 THE EUCLIDEAN DISTANCE RATIO FOR THREE DATA INSTANCE IN EXAMPLE 1

Data instance	T_1	T_2	T_3
T_1	0	0.359	0.112
T_2	0.359	0	0.445
T_3	0.112	0.445	0

A similarity calculation method proposed by Gower [12] to measure the similarity ratio between two objects with missing entities. Gower's method (S_{15} as denoted by Legendre et al. in [25]) has the flexibility to ignore the comparison for resemblance data with missing portion. That is, if patient's data in a particular instance does not have all measurements, the Gower's method (S_{15}) could compute the similarity [12] by activating an additional coefficient (called *Gower's coefficient*) to turn on or off the yielded sub-similarity ratio (or sub-distance ration). The Gower's coefficient (denoted *w*) is carrying a value of 1 if the entity (in data instance) is existing in both instances, otherwise 0 if the entity is missing (or sometimes null value) in both instance or in either of them. The coefficient is then multiplied by the sub-similarity ratio between entities as in the below Eq. (3).

$$S_{15}(\mathrm{x}, \mathrm{y}) = \frac{\sum_{i=1}^{n} w_i \mathrm{gS}(\mathrm{x}_i, \mathrm{y}_i)}{\sum_{i=1}^{n} w_i} \tag{3}$$

Where S_{15} is the similarity calculation proposed by Gower, w_i is the coefficient with 0 or 1 value according to the existing of entities, and *n* is number of entities in the data instance records. Finally, *S(x,y)* is the original similarity methods by the Euclidean distance equations (refer to Eqs. 1 and 2). Now, according to Gower's method, Table 4 is listing similarity ratios

DOI: 10.5963/LSMR0402001

computed. Notice that Table 4 is showing two numbers, the one in parentheses are the similarity ($S=1-D_{norm}$) while the other number is the distance. From Table 4, we can clearly notice that the distance ratio between the data instance T_1 and T_2 (0.033) are higher than the distance between T_2 and T_3 (0.015). Therefore, Gower's method was able to match accurately the similarity (or distance) ratio among instances while there are missing portion of the data records. That means T_2 and T_3 are the right instances-pair to be cluster first.

TABLE 4 THE GOWER SIMILARITY RATIO (IN PARENTHESIS) FOR THREE DATA INSTANCE IN EXAMPLE 1

Data instance	T_1	T_2	T_3
T_1	0 (1)	0.033 (0.967)	0.061 (0.939)
T_2	0.033 (0.967)	0 (1)	0.015 (0.985)
T_3	0.061 (0.939)	0.015 (0.985)	0 (1)

The challenge of comparing data instances while there are portions of the data missing has been not considered in existing works such as those referred in this paper. As we illustrated in the above example 1, Gower's similarity method is proofing the capability of evaluating data instances while portion of the data is missing. For that reason, to visualize a time-based patient's history, we have to consider the challenge that portion of the data could be missing, and consequently we must use methods workable with this condition. To address this clearly, if we review the closest instances-pair in Table 3, we find that T_1 and T_3 are the closest pair among all combinations (with distance 0.112). However, they must not be the closest as they are the most distinct pairs in the table. This ambiguity is handled smoothly be Gower's method (see Table 4). The distance between T_1 and T_3 (0.061) is the highest among all distances in the table.

B. Clustering Algorithm

There are many available clustering algorithms that could do the clustering task for patient's history. Legendre et al. [25] has classified clustering algorithms as hierarchical and non-hierarchical clustering. Moreover, authors further explained the partitioning and agglomerative clustering. However, in this subsection, we are focusing our review sole on agglomerative clustering algorithms.

Agglomerative clustering algorithm is a hierarchical grouping algorithm that combines individual objects and sets according to similarity measures and grouping threshold (called *alpha* or α). If there are two individual objects want to combine in single set (or grouped), the agglomerative algorithm is simple. The similarity ratio between the two objects are calculated and evaluated by the threshold for grouping permission. However, if there are two sets of objects (say set A and B) or object (a) and set (B) want to combine, then the similarity ratios are calculated between objects located at set A and objects located at set B. similarly, for the object (a) which wants to combine with set B, the similarity ratios are calculated between object a and objects located at set B. This example is illustrated in Fig. 1.

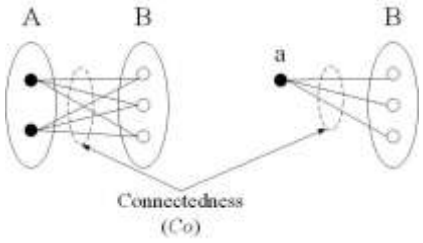

Fig. 1 Two sets A and B (left) or object "a" and set B (right) would be combined if the connections between their pair objects are satisfying the similarity ratio. The connectedness between the two sets (or the object and set) is percentage of successful similar pair to the all pair combinations.

Agglomerative clustering algorithms are three types of algorithms that define the overall similarity between two sets of objects (or object and set of objects). These algorithms are namely single linkage [32], complete linkage [33], and intermediated linkage and average linkage.

Single linkage algorithm [32] suggests if there is one pair of objects among objects pair between the two sets A and B (or between object a and any object at set B) that satisfy the similarity ratio and the grouping threshold, then these two sets (A and B) are permitted to be combined with (or object a is permitted to join set B). This algorithm is a loosely clustering algorithm which requires any single link between objects to permit the grouping. Therefore it is not adequate for clustering patient's historical data.

Complete linkage algorithm [33] suggests that all pairs of objects among the two sets A and B (or all pair between object (a) and objects at set B) must satisfy the similarity ratio and the threshold to permit grouping, otherwise they are not permitted to group. The complete linkage algorithm is demanding a 100% similarity condition among all clustered objects. This condition is much accurate compared to single linkage algorithm, however, in patient's historical data will have many clustered (subsets) data which is difficult to meet this requirement.

Average linkage algorithm suggests an update to the complete linkage algorithm, which is a 50% of pairs satisfying the similarity ratio and grouping threshold will qualify the two sets A and B to combine (or object a to join set B). The average

DOI: 10.5963/LSMR0402001

linkage algorithm is enabling another threshold to permit the grouping of objects and sets among data. This threshold is known as the connectedness between sets (*Co*). In this algorithm (average linkage algorithm) the connectedness is 50% while it was 100% in the complete linkage algorithm and 1/n % in single linkage algorithm (n is the number of pairs among objects). The average linkage algorithm is divided further into four methods according to the metric properties of the averaging technic. We provided the two naming styles so interested readers could distinguish them in literature.

- Unweighted arithmetic average clustering or Unweighted Pair-Group Method using Arithmetic averages (UPGMA) [34]

  This algorithm is checking the similarity ratio among objects, and the highest will be grouped. Then, the algorithm considered these two objects as one, and the similarity rations between these two objects and the remaining objects will be averaged. This will results that two objects will be removed and replaced with a single object, and the similarity ratio of this object with any objects will be the average of the two previous similarity ratios. However, if this algorithm is keep checking the similarity ration, there could be a chance to find the highest similarity ratio between one (real) object and one (re-calculated) object. In this case, the procedure takes place and it means the two objects (real and re-calculated) will be removed and replaced with new object and the similarity ratio of other objects will be averaged without giving a higher weight to the similarity ratio of the re-calculated object (as it is representing two objects). From here, we have illustrated the "unweighted arithmetic" and the "average" terms in the algorithm title.

- Weighted arithmetic average clustering or Weighted Pair-Group Method using Arithmetic averages (WPGMA) [35]

  Similarly, this algorithm is checking the similarity ratio among objects, and the highest will be grouped. Then, the algorithm considered these two objects as one, and the similarity rations between these two objects and the remaining objects will be averaged. However, if the situation of finding the highest similarity ratio is between one (real) object and one (re-calculated) object, then the procedure is as follows. The similarity ratio of other objects will be arithmetically averaged by adding a higher weight to the similarity ratio of the re-calculated object (as it is representing two objects). The weight is (in this case) the number of the original objects represented by these objects. So, if there is two re-calculated objects to be grouped, then each similarity ratio would be multiplied by the original number of the represented objects. And that is why we have the "weighted" term in algorithm title.

- Unweighted centroid clustering or Unweighted Pair-Group Method using Centroid (UPGMC) [36, 37]

  The algorithm is similar to the previous UPGMA, however, it differs in the new representing object (after grouping). The similarity ratios of the new object are re-computed geometrically to find a centroid location of the new object. And accordingly, the similarity ratios are re-produced according to that geometric location.

- Weighted centroid clustering or Weighted Pair-Group Method using Centroid (WPGMC) [38]

  Lastly but not least, the Weighted Pair-Group Method using Centroid is similar to UPGMC but with enhancement. That is related to averaging the similarity ratios of the new centroid object. So instead of calculating the similarity directly, a weighing system method (as in WPGMA) is added.

Among these algorithms, we used the weighted arithmetic average clustering (WPGMA) as our based grouping algorithm but other algorithms are useful too and GIDS user can choose among them during the implementation of GIDS workflow.

In this paper, we combined the similarity method suggested by Gower and the grouping algorithm (WPGMA) proposed by Sokal et al., and we followed the chronological cluster algorithm proposed by Legendre et al. [26, 27] to produce the visualization of patient's history. The chronological clustering is a hierarchical clustering that groups resemblance data in an agglomerative way. The algorithm is agglomerative (not partitioning) where the data entities are initially individuals. The algorithm start by combining the adjacent-in-time entities repeatedly until a threshold (alpha) is reached. The key attractive feature of this algorithm is that it has the broad-scale and finer-scale steps, which are used to overview the data and identify detailed observations, respectively. The broad- and finer-scale steps in chronological clustering differ from the lens style in [3].

IV. GRID-BASED INTERACTIVE DIABETES SYSTEM

Our previous work in [23, 24] addressed the illustration of the Grid-based Interactive Diabetes System (GIDS). However, in this section we will provide basic understanding of the GIDS. GIDS is a web-enabled Grid-based application that uses the distributed computing technology to perform bioinformatics analysis application for diabetes disease [23]. GIDS help medical doctors and scientists who are interested in finding the relationship between diabetes condition of a patient and his/her lifestyle. GIDS is the view optimizer. For example, it can produce several views of patient's historical information. GIDS frees medical doctors from the burden of reviewing patient's medical records such as those reported by laboratory tests, radiology x-rays images (PACS) and cardiology ECG analysis. Putting all these technologies together in single environment and reporting intelligently the interesting changes to medical doctor is a challenge. Therefore, based on Legendre, P. et al. chronological clustering algorithm [25-27], GIDS helps to display the interesting changes into an understandable chart graphs e.g. blood sugar that easily helped in the study [23].

DOI: 10.5963/LSMR0402001

V. WORKFLOW STRUCTURE

Patient's history varied on both types and frequencies, and GIDS needs a workflow system to process the study of such diversity. Therefore, we constructed a web-enabled workflow initializer that interacts with user (doctors or medical researchers) via outlet called output. Fig. 2 is illustrating the complete structure.

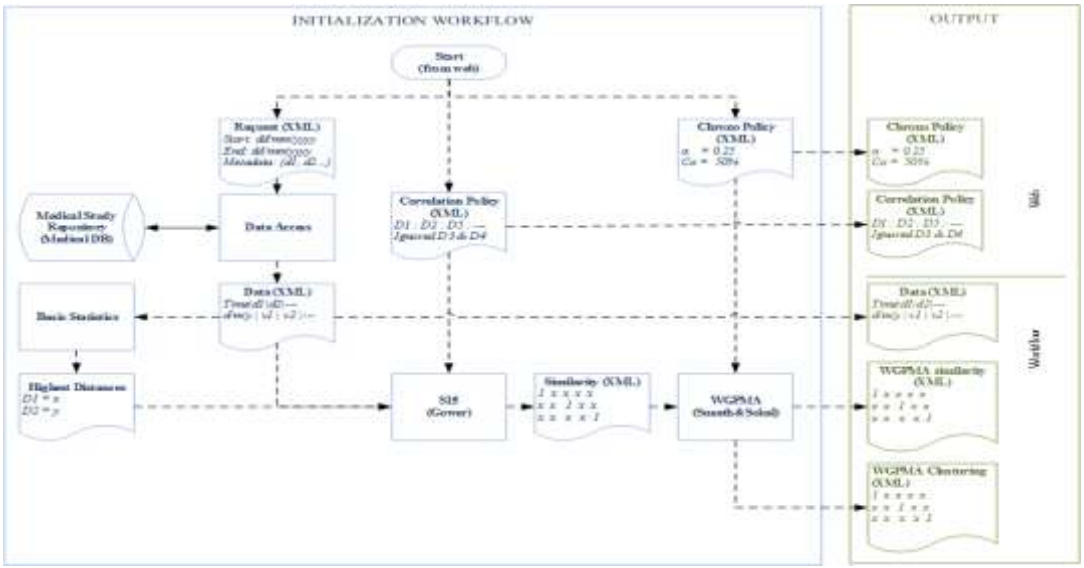

Fig. 2 Workflow structure

A. Initialization Workflow

The initialization workflow consists of three parallel tasks namely requesting data, correlation policy, and chronological policy (or viewer setting). The first task (requesting data) is a simple data acquisition from medical database. Then, it followed with basic statistics analysis such as finding the highest distance (or variations) among data. The outcome goes to a similarity analyser (in this work we used Gower (S_{15}) as in [12]). However, to do this similarity subtask, a correlation policy is required. The correlation policy task is defining which data to be engaged in the similarity calculation. For example, doctor is interested in blood sugar measurements to display but he/she requires engaging the variation of blood pressure or cholesterol measurements in the analysis of patient's history. The yield similarity data used by other subtask called the weigh-group average (WPGMA) which is a weighing task for grouping data [35]. Finally, the chronological policy is responsible to control the agglomerative algorithm parameters, which are the tuners for optimized views. The chronological policy is online controller that could be interactively (online) changed by the user and the results will be feedback in short time (depending on the backend performance computing system). All data and controls are moving across the workflow in a XML data files.

Lastly, the key element of having workflow to support the timeline visualization is that interested users who wish to switch to different similarity algorithms or clustering methodology are able to replace them. For example, if the Gower's (S_{15}) similarity method is to be replaced by the Euclidean distance, then user could replace the compositor component of Gower in the workflow with the Euclidean distance (as long as the output is converted from distance ratios to similarity ratios as in Eq. (1). Similarly, the clustering method WPGMA could be replaced with other methods such as WPGMC, UPGMA, Single-linkage, Complete-linkage, and k-mean (with adequate revised version).

B. Output

The web-enabled outlet of our workflow with GIDS is the output. It is the place for user to update the tasks according to chronological policy and correlation policy, and visualized the data. However, we enable the view of the outcome from the WPGMA subtask, so the user of the system will see the hidden computation happened and adjust the correlation policy accordingly.

VI. EVALUATION

Our main goal of the evaluation is to examine the system capability of producing different views of patient's data. Therefore, we tested GIDS with a diabetic patient's data (less than a month data set). We captured the views generated for different displays, in particular, tide display, desktop monitor and smart phone display. The goal of this test was to display the full history of a patient's blood sugar into different displayers, and compare the effects of variation to the available display limitations.

A. Test-Bed

DOI: 10.5963/LSMR0402001

GIDS is tested in small size computer lab to evaluate its usability and functionality. The lab consists of three physical computers each was 3.2 GHz CPU and one laptop used as user's PC to access GIDS Portal website. GIDS system was installed into one computer and Kepler was installed into the same computer. Because there was a need of having multiple of computers work as Condor-based Grid resources and Rocks Cluster [38] was installed into other two computers as one master node and one slave node. Three virtual machines were created on top of Rocks Cluster, and each with Condor enabled.

B. Patient's Data

A diabetes study workspace was created for anonymous patient, who was under supervision for 27 days starting from 1st of January, 2010, and a blood sugar test was taken daily. Two adjustments were applied to the study to examine the usability of the system to help medical researches. First, a study workspace was created; then an adjustment was applied on the Chronological Clustering Algorithm, after which a change was applied to time domain of the supervision period.

C. Results

Our solution was able to provide an optimized summery of patient history without hiding major variations. Fig. 3 shows three type of diabetes trend for three types of display, namely big screen (such as tide display), normal personal computer (PC) monitor and smart device (such as smart phone). Each image shows the same trend of blood sugar and all of them displayed the major variation of blood sugar in 18th and 19th of January (192 and 189 mg/dl respectively). However, minor changes are hiding in PC monitor and smart phone as they are not possible for human eye to interpret with limited displayer.

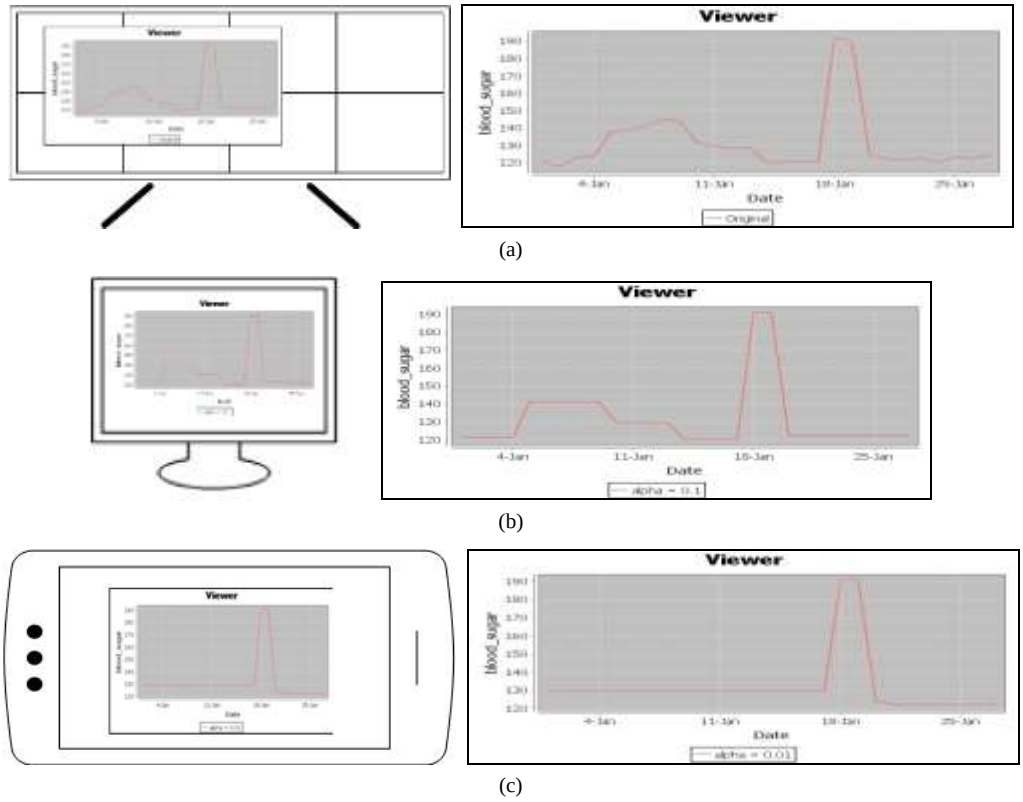

Fig. 3 Different views of patient's blood sugar measurements: (a) Original data onto big screen or tide displayer; (b) Optimized view for personal computer's displayers or monitor; (c) Optimized view for smart devices with limited displayer such as smart phones

In this case, if doctor is interested in observing any minor variation in the trend, doctor could further zoom in the trend (not the figure) by re-adjusting the interested period. To comply with this need, we examine the functionality of zooming into portion of displayed data for better understanding in patient's condition. Fig. 4 illustrates that the functionality of zooming could show minor hiding data variations. I.e. the initial view of PC's monitor (Fig. 4 top) shows the blood sugar rose up on January 5th (140 mg/dl), and then it decreases gradually until January 14th (120 mg/dl where it reaches its initial condition). The interested part of the view (in here) is the period started on January 5th and ended on January 14th. After re-adjusting the period and with the help of the zooming functionality in the system, we were able to see more details in that period. Further, we have noticed that the blood sugar did not rise up once on January 5th (140 mg/dl) but the hidden rising in blood sugar is on Jan. 8th and 9th. Using this zooming functionality, doctor will be able to interpret any slight changes in patient's condition. Currently, we are developing this functionality to report these minor changes (as small dots or stars) in the graphical trends. So doctors may notice any hidden minor variation of data and if he demands to see the details, the system could visualize them.

DOI: 10.5963/LSMR0402001

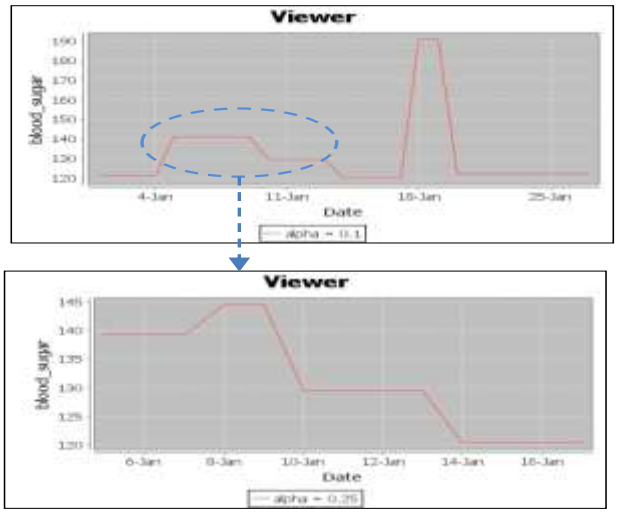

Fig. 4 A zooming in example to view interested changes in patient's data between January 5th and 16th

VII. CONCLUSIONS

The main goal from the GIDS is assisting medical researchers who are interested in studying diabetes disease. GIDS enables a broader view of diabetes patient's condition and predicts the future implication based on current developments in some diabetes factors such as blood sugar or fat. GIDS uses the Chronological Clustering Algorithm for analysis diabetes records. However, it is expandable to cover any chronic diseases that have similar motivation as the diabetes. Our main future work is the migration of the Grid-based computing backend infrastructure to a Mobile Cloud Computing.

REFERENCES

[1] A. A. T. Bui, D. R. Aberle and Hooshang Kangarloo, "TimeLine: visualizing integerated patient records," Information Technology in Biomedicine, IEEE Transactions, vol. 11(4), pp. 462-473, July 2007.

[2] Combi Carlo and Barbara Oliboni, "Visually defining and querying consistent multi-granular clinical temporal abstractions," Artificial intelligence in medicine, vol. 54(2), pp. 75-101, 2012.

[3] Rhys Tague, Maeder Anthony and Quang Vinh Nguyen, "Interactive visualisation of time-based vital signs," Advances in Visual Computing, Springer Berlin Heidelberg, pp. 545-553, 2010.

[4] S. Toyoda, N. Niki, H. Nishitani, "SAKURA-Viewer: intelligent order history viewer based on two-viewerpoint architecture," Information Technology in Biomedicine, IEEE Transactions, vol. 11(2), pp. 141-152, March 2007.

[5] Toyoda Shuichi and Noboru Niki, "Information visualization for chronic patient's data," Information Search, Integration and Personalization, Springer Berlin Heidelberg, pp. 81-90, 2013.

[6] Malik Muhammad Sheraz Arshad and Suziah Sulaiman, "Analytical comparison of factors affecting EHR visualization for physicians in public health care units," Advances in Visual Informatics. Springer International Publishing, pp. 605-614, 2013.

[7] Alexandra Pomares-Quimbaya, et al., "Improving decision-making for clinical research and health administration," Engineering and Management of IT-based Service Systems, Springer Berlin Heidelberg, pp. 179-200, 2014.

[8] William Hsu, et al., "Context-based electronic health record: toward patient specific healthcare," IEEE Transactions on Information Technology in Biomedicine, vol. 16(2), pp. 228-234, 2012.

[9] Lucio Grandinetti and Ornella Pisacane, "Web based prediction for diabetes treatment, Future Generation Computer Systems," vol. 27(2), pp. 139-147, February 2011. ISSN 0167-739X, http://dx.doi.org/10.1016/j.future.2010.08.001.

[10] Brzostowski Krzysztof, Jarosław Drapała and Jerzy Świątek, "System analysis techniques in eHealth systems: a case study," Intelligent Information and Database Systems, Springer Berlin Heidelberg, pp. 74-85, 2012.

[11] Ziesche Soenke and Sahar Motallebi, "Personalized remotely monitored healthcare in low-income countries through ambient intelligence," Evolving Ambient Intelligence, Springer International Publishing, pp. 196-204, 2013.

[12] John C. Gower, "A general coefficient of similarity and some of its properties," Biometrics, pp. 857-871, 1971.

[13] Scott M. Pappada, Brent D. Cameron, Paul M. Rosman, Raymond E. Bourey, Thomas J. Papadimos, William Olorunto and Marilyn J. Borst, "Neural network-based real-time prediction of glucose in patients with insulin-dependent diabetes," Diabetes technology & therapeutics, vol. 13(2), pp. 135-141, 2011.

[14] Carmen Pérez-Gandía, A. Facchinetti, G. Sparacino, C. Cobelli, E. J. Gómez, M. Rigla, A. De Leiva and M. E. Hernando, "Artificial neural network algorithm for online glucose prediction from continuous glucose monitoring," Diabetes technology & therapeutics, vol. 12(1), pp. 81-88, 2010.

[15] Chernetsov Serge, Anatoly Karpenko, and Alexander Trofimov, "Neural network-based blood gucose control system for type I diabetes patients," International Journal of Life Science and Medical Research, vol. 2(2), pp. 15-18, June 2013. DOI: 10.5963/LSMR0202001

[16] Elena Daskalaki, Aikaterini Prountzou, Peter Diem and Stavroula G. Mougiakakou, "Real-time adaptive models for the personalized prediction of glycemic profile in type 1 diabetes patients," Diabetes technology & therapeutics, vol. 14(2), pp. 168-174, 2012.

DOI: 10.5963/LSMR0402001

[17] I. Altintas, C. Berkley, E. Jaeger, M. Jones, B. Ludascher and S. Mock, "Kepler: an extensible system for design and execution of scientific workflows," 16th International Conference on Scientific and Statistical Database Management, Proceedings, 2004, pp. 423-424.

[18] Ewa Deelman, James Blythe, Yolanda Gil, Carl Kesselman, Gaurang Mehta, Sonal Patil, Mei-Hui Su, Karan Vahi and Miron Livny, "Pegasus: Mapping Scientific Workflows onto the Grid ," Across Grids Conference 2004, Nicosia, Cyprus.

[19] Peter Couvares, Tevik Kosar, Alain Roy, Jeff Weber and Kent Wenger, "Workflow in Condor," in Workflows for e-Science, I. Taylor, E. Deelman, D. Gannon, M. Shields, Ed. Springer Press, January 2007 (ISBN: 1-84628-519-4)

[20] S. R. Amedolia and F. Estrella, "MammoGrid: A Service Oriented Architecture based Medical Grid Application," Proceedings of the 3rd International Conference on Grid and Cooperative Computing, Wuhan, China, 2004.

[21] N. Chrisochoides, A. Fedorov, A. Kot, N. Archip, P. Black, O. Clatz, A. Golby, R. Kikinis and S. Warfield, "Toward Real-Time Image Guided Neurosurgery Using Distributed and Grid Computing," Proceedings of the 2006 ACM/IEEE, SC\06 Conference (SC'06.)

[22] L. Srinivasan and J. Treadwell, "An Overview of Service Oriented Architecture, Web Services and Grid Computing," http://h71028.www7.hp.com/ERC/downloads/SOA-Grid-HP-WhitePaper.pdf

[23] F. Al-Hazemi, et al., "Grid-Based Interactive Diabetes System," First IEEE International Conference on Healthcare Informatics, Imaging and Systems Biology (HISB), 2011, pp. 258-263

[24] Fawaz AL-Hazemi, "Grid-based Workflow System for Chronic Disease Study," In Proceedings of the 4th FTRA International Conference on Advanced IT, engineering and Management (FTRA ACS-14), 2014, pp. 162-163.

[25] Legendre Pierre and Louis Legendre, Numerical Ecology, vol. 20, 2012, Elsevier.

[26] P. Legendre, S. Dallot and L. Legendre, "Succession of Species within A Community: Chronological Clustering, with Applications to Marine and Freshwater Zooplankton," Am. Nat. I. vol. 125, pp. 257-288, 1985.

[27] M. Bell and P. Legendre, "Multicharacter Chronological Clustering in A Sequence of Fossil Sticklebacks," Syst. Zool., vol. 36(1), pp. 52-61, 1987.

[28] American Diabetes Association, "Standards of medical care in diabetes–2007," Diabetes Care, vol. 30(Suppl 1), pp. S4-S41, 2007. DOI: 10.2337/dc07-S004

[29] Keogh, Eamonn and Jessica Lin, "Clustering of time-series subsequences is meaningless: implications for previous and future research," Knowledge and information systems, vol. 8(2), pp.154-177, 2005.

[30] Jessica Lin, Keogh Eamonn and Truppel Wagner, "Clustering of streaming time series is meaningless," Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery, ACM, 2003.

[31] Esling, Philippe and Carlos Agon, "Time-series data mining," ACM Computing Surveys (CSUR), vol. 45(1), p.12, 2012.

[32] Glenn W. Milligan, "Clustering validation: results and implications for applied analyses," Clustering and classification, vol. 1, pp. 341-375, 1996.

[33] Sørensen Thorvald, "A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons," Biol. Skr., vol. 5, pp. 1-34, 1948.

[34] Peter H. A. Sneath and Robert R. Sokal, Numerical Taxonomy, The principles and practice of numerical classification, 1973.

[35] Robert R. Sokal and Charles D. Michener, A statistical method for evaluating systematic relationships, University of Kansas, 1958.

[36] Godfrey N. Lance and William Thomas Williams, "A general theory of classificatory sorting strategies 1. Hierarchical systems," The Computer Journal, vol. 9(4), pp. 373-380, 1967.

[37] John C. Gower, "A comparison of some methods of cluster analysis," Biometrics, pp. 623-637, 1967.

[38] Federico D. Sacerdoti, Sandeep Chandra and Karan Bhatia, "Grid Systems Deployment & Management Using Rocks", IEEE International Conference on Cluster Computing, September 2004, San Diego.

Fawaz AL-Hazemi received his B.S. degree in Computer Engineering from KFUPM, Saudi Arabia in 2003 and his M.S. degree in Information and Communications Engineering from KAIST, South Korea, in 2010. He is pursuing his PhD in Electrical Engineering at KAIST. His research interests are data center management, grid and cloud computing, and healthcare systems.

Mr. AL-Hazemi is a member of IEEE and ACM. He is member of Technical Program Committee (TPC) for several international conferences as well as reviewer in others international conferences. He published more than 15 technical papers, and he was the recipient of FTRA AIM 2014 "Best Paper Award".

DOI: 10.5963/LSMR0402001