CHEERUP: A General Software-Environment for Building, Using and Administering Predictive Monitoring Portals

Silvano MUSSI
Member of AICA – Associazione Italiana per l'Informatica ed il Calcolo Automatico
Piazza Rodolfo Morandi 2, 20121 Milano, Italy
mussisilvano@tiscali.it

Abstract—**The intended meaning of the term "predictive monitoring" used in the paper is the following. A population of subjects (living beings, machines, works of art, etc.) is monitored by a domain expert with regard to the possible occurrence of an undesired/desired event E. More precisely, an expert periodically (e.g. every two years, every week, etc. depending on the specific application) examines the single subjects and, for each of them, enters examination outcomes in a database where statistical data are automatically processed in order to produce probabilistic inferences about the occurrence in the future of E for the subject under examination (individualized prediction). This allows the expert to take suitable measures in advance in order to prevent/favour the occurrence of E for the subject. Such an approach to predictive monitoring requires that the expert who monitors subjects has at his/her disposal a suitable software system provided with database and algorithms for both properly managing monitoring-processes and producing probabilistic predictions. The paper presents CHEERUP : a prototype product, usable via Internet, that consists in a general software-environment for building, using and administering specific predictive monitoring software-systems (in the paper called portals).**

***Index Terms*—computer applications, predictive models, learning systems, data processing.**

I. Introduction

The great number of works concerning predictive monitoring, published in scientific journals and conferences both in past and in recent years, gives evidence of both the modernity of the topic and the remarkable effort so far accomplished by the researchers community. Literature shows that, in general, prediction has been intended in the sense of prevention, that is as a means for preventing undesired events. Actually the possibility of getting early warnings before an undesired event may occur has always been very appealing. Let us think, for example, of prevention of high risk events for health, or serious faults or anomalies of costly and strategic industrial equipments or plants. In the proposal, prediction regards both preventing undesired events and favouring desired events. The paper presents a general software environment for building and using application-oriented predictive monitoring tools. Let us notice that the habitual use, in the long period and in various application fields, of predictive monitoring tools has also a cultural side-effect, it generates a world outlook consisting in facing problems with an underlying look-ahead attitude: you cannot effectively face the present regardless of the future.

A. What CHEERUP consists in

The intended meaning of the term "software-environment" used in the paper is the following: a software-environment is a set of software functions provided to a user to reach a certain goal (for example, a word processing environment is a collection of software functions provided to a user to create a document: layout functions, character typographic style functions, etc.). If the software-environment is usable via Internet, it is called Web-environment. CHEERUP is a Web-environment addressing users having the goal of monitoring subjects of a given population in order to prevent/favour, for each of them, the occurrence of an undesired/desired event. CHEERUP provides users with several functions that make it easy and simple to build, use and administer software tools (in the paper called predictive monitoring portals, or, for short, simply: portals) for monitoring subjects and producing individualized probabilistic predictions about the future occurrence (for each subject being monitored) of a certain undesired/desired event.

Many real world domains are characterized by the following paradigm. There is a population of subjects (human beings, machines, etc.). There is an event E (undesired or desired) that may happen or not to each subject of the population. The occurrence probability of E for a subject may be affected by both the mere aging of the subject and the contexts (i.e. conditions) in which the subject ages. The subjects are monitored at constant time intervals by a domain expert. During the monitoring session of a subject the expert inserts (into the database) both the presence/absence (for the subject) of the contexts and the fact "E has occurred/not-occurred" (for the subject). In case E = not-occurred, the purpose of monitoring is having a probabilistic prediction about E occurrence in the future for the subject being monitored, so to be able to take suitable measures in advance. CHEERUP may be suitably used for any problem that can be represented as an instance of that paradigm. As a consequence the number of possible CHEERUP applications is great indeed. They may be distinguished in two major categories: if E is undesired we have to do with numerous possible applications concerning prevention (e.g. preventing ictus, preventing car-engine fault, etc.), if E is desired we have to do with applications in which we want to favour the occurrence of E (e.g. in the

Digital Object Identifier 10.4316/AECE.2011.04010

sport domain the setting up a certain record is a desired event for athletes, in the education domain the passing a certain exam is a desired event for students, etc). CHEERUP applies to a great number of heterogeneous domains (Education, Sport, Cultural heritage, Environment, Medicine, Natural Sciences, Social Sciences, Industrial Technology, Economy).

CHEERUP is a general environment which is in turn structured in target environments: the environment for building portals for specific applications, for short, the *Portals Building environment*; the environment for using portals, for short, the *Portals Using environment*; the environment for administering portals, for short, the *Portals Administering environment*; the environment for administering subjects, for short, the *Subjects Administering environment*; the environment for administering the environments (i.e. the four cited environments), for short, the *Environments Administering environment*.

CHEERUP supports decision making by means of probabilistic prediction and simulation. For each subject being monitored the expert can get an answer to the question: if the subject kept on staying, even in the future, in the same contexts in which it is at the present time, how much in the future the occurrence probability of E would be for the subject? Knowing that, may help choose in advance the right measure to prevent/favour the occurrence of E for the subject. Moreover it is possible to simulate a situation of contexts different from the real one, and calculate its probabilistic consequences in the future. Knowing that, supports the right choice among possible alternative measures aiming to prevent/favour the occurrence of E for the subject.

CHEERUP facilitates co-operation among work-groups, providing several facilities useful to work in team, in a structured organization. In order to discipline co-operative work, distinct roles are defined (and regulated by authorization levels) for using the related environments: the portal builder role, the portal user role, the portal administrator role, the subjects administrator role, the super-administrator role (i.e. the role of administering the environments).

CHEERUP is a product easy to use. It is equipped with numerous check functions about correctness and coherence of data entered by users, and provides many functions for making its use easy, friendly and proper. Its installation is simple and guided step by step. Database tables are automatically created. It is written in Asp and uses the database management system Mysql. CHEERUP has been conceived and carried out by the author of this paper.

B. Paper organization

The following of the paper is organized in the following way. Section II presents the Portals Building environment by illustrating its main functions at a conceptual level, giving emphasis to a major strength-point of CHEERUP: the easiness by which a portal builder equips its portal with the basic infrastructures that will then be used in the Portals Using environment. Section III presents the Portals Using environments by conceptually illustrating the central features of CHEERUP: monitoring subjects, predicting and explaining predictions, learning to improve predictions. Section IV presents the three administering environments: the Portals Administering environment, the Subjects Administering environment, the Environments Administering environment. Section V discusses the proposal in the context of related work. Finally, section VI draws some conclusions.

II. The Portals Building Environment

In order to make the presentation easier to read and understand, let us make abstract concepts concrete by referring to an example chosen inside a specific domain: the medicine domain. However the reader should not intend that CHEERUP is a proposal suitable to medicine only, the medicine domain is considered just as an example. The example is not developed in a scientific rigorous manner as a physician would do (the terminology too may be imperfect). It appears incomplete, very simplified and/or naive, especially if the reader is a physician. The purpose of such an example is to make even a non physician reader able to get a clear comprehension of how CHEERUP works.

The medicine domain is a typical case in which there are several undesired events whose occurrence is favoured if the subject passes long time in some contexts that are commonly called risk factors. For example, let us consider the event *First Cardiac Infarct* (the undesired event E). Among the set of the related risk factors we might identify: *obesity*, *smoke, hypertension*, *abnormal cholesterol levels*, etc. Let us suppose that a medical monitoring enterprise to prevent the occurrence of the first infarct has been put into practice for a population of mail subjects starting from a certain age. During a monitoring session the physician takes note of the presence/absence of the considered risk factors. Let us notice though that for some risk factors it might not be enough to know that at session time they turn out to be present for the subject under examination. In fact it might also be necessary to know how long the subject has passed in presence of those risk factors. For example, the longer the subject has been smoking the higher the contribution that smoke gives to rise the occurrence probability of the event *First Cardiac Infarct* is. After this premise let us start to build the *First Cardiac Infarct* portal (in CHEERUP, portals have the same names as the related undesired/desired events). The home-page of the Portal Building environment consists in a set of functions that allow the builder to create and edit the various components of the portal. Let us examine some main functions.

A. The function: Edit monitoring parameters

The function “Edit monitoring parameters” allows the builder to define: the time-unit (year, month, week, day) that will be used by the new portal, the initial age (the subject age at which the first monitoring session should start), the (constant) time interval between a monitoring session and the next one, the total number of monitoring sessions, the minimum number of cases (threshold value) needed to make probabilistic inferences. For example, let us suppose that in the *First Cardiac Infarct* example the selected time-unit is *year*, the *initial age = 60 years*, with *time interval between monitoring sessions = 2 years* and *total number of sessions*

= 6 (so that the last monitoring session occurs when subject age = 70 years).

B. The function: Edit contexts

The function "Edit contexts" leads the builder to a form page in which he/she defines possible contexts in which a subject might be (for example, the familial anamnesis *genetic predisposition*) or in which a subject can elapse time, knowing that elapsing a significant amount of time in one or more of such contexts affects E occurrence probability (for example, elapsing long time in the contexts (risk factors): *obesity*, *smoke, hypertension*, can condition the occurrence probability of *First Cardiac Infarct*).

C. The function: Edit context states

The function "Edit context states" allows the builder to select a context C among the ones defined and enter a set of C states. A context C is like a variable and a state of C is like a value that can be assigned to C. Some contexts may be qualified by the builder as fixed-state contexts. Fixed-state context means that in all the sessions following the first one the context must have the same state it has in the first session. For example, a historic anamnesis fact concerning *genetic predisposition* does not change in the course of the monitoring sessions. As a consequence the risk factor *genetic predisposition* is a typical fixed-state context. Let us suppose that for such risk factor the builder enters the set of states {*no , yes*}. Vice versa, as for *obesity* the builder might need to define a more articulate set of states. For example, instead of a simple state *yes* the builder might make a distinction between light and important obesity. So he/she might define the state *yes-light* and the state *yes-important*. Given the relevance of this last state, the builder might retain that it would be important to specify how long the subject has elapsed in the context *obesity* with state *yes-important*. To provide the user of the portal with such a possibility, the builder assigns the state the qualification of "time-sensitive state". The concept of time-sensitive state plays a relevant role in CHEERUP. What is a time sensitive-state? Assigning the time-sensitive qualification to a state S of a context C means that we want that the text of the state explicitly includes how long the subject has elapsed in the state S of the context C. In order to accomplish that, the state text is automatically extended with the string (called temporal-part): "and such state has lasted for <N time-unit>", where N is an integer number ranging from 0 to the final age considered in the monitoring process. So we have a state text that is compound of a static part plus a dynamic temporal part. For this reason, states that need to be completed with temporal part in order to be fully significant are called time-sensitive states. Turning back to the example, the state text *yes-important* is extended with the addition of the temporal part: *and such state has lasted for* N *years*, where N ranges from 0 to 70 (i.e. the final age considered in the *First Cardiac Infarct* monitoring process). In the Portal Building environment a builder qualifies a state as time-sensitive by simply selecting the time-sensitive flag beside the state text description. It will be up to the Portal Using environment to compose, when needed, a proper compound state.

D. The function: Portal ready to definitely enter the Using environment

At the end of the building work, the new portal is like a new ship ready to leave the shipyard, enter the sea and be used by the crew in the service of passengers. The transfer from the shipyard to the sea is carried out by launching the ship. Similarly, at the end of the building work, the new portal is ready to leave the Building environment, enter the Using environment and be used by domain experts in the service of subjects. The transfer from the Building to the Using environment is carried out by the function "Portal ready to definitely enter the Using environment". Such transfer takes place through tree steps. Step 1: a view of the whole portal is shown to the builder to receive his/her confirmation that everything is OK. If step 1 is OK, then step 2 takes place: correctness and coherence checks are carried out all over the portal. Finally, if everything turns out to be OK, the builder is asked for the last time if he/she confirms the decision to carry out the transfer. If the builder confirms, step 3 takes place: the set of database tables of the new portal are created in the Using environment; the information (contexts, states, etc.) collected in the Building environment are copied into the new database tables of the Using environment; all the database tables created and used during the portal building process are eliminated from the Building environment.

E. The function: Portal ready to be tested in the Using environment

The Building environment provides the builder with another very useful option: the function "Portal ready to be tested in the Using environment". Such function inserts the portal into the Using environment in TEST mode. In this case the new portal enters the Using environment, but the database tables of the portal in the Building environment are not eliminated. If it is necessary, the builder has the possibility to go back to the Building environment in order to suitable modify the portal. If the builder selects the TEST mode transfer, the Using environment, in order to make the portal usable in all its functions, simulates that, for any combination of context states, in any session the portal has already collected a number of cases greater than the threshold value required to make probabilistic inferences. As a consequence the portal user can test how the new portal works and looks in the Using environment, knowing, of course, that probabilistic values showed during testing are dummy. The user of the new portal in order to be able to test it has to monitor some subjects, subjects that, for the sake of clarity have to be treated separately from the ones already present in the subjects database. As a consequence the Using environment automatically provides a portal used in TEST mode with ten dummy subjects so that the portal test is accomplished by monitoring dummy subjects only.

III. THE PORTALS USING ENVIRONMENT

The Using environment is the one that has to do with the ultimate purpose of CHEERUP: monitoring and predicting.

A typical monitoring session for a subject is structured in four basic steps: acquisition of subject data, probabilistic inference about the future occurrence (for the subject) of the undesired/desired event, presentation of probabilistic prediction, session termination and learning. In the following each step is presented in a detailed way, and we will play the role of portal user. However, before starting to describe the single steps let us examine the basic model underlying the whole proposal.

A. Basic model

The basic theoretic model used by CHEERUP is the Dynamic Bayesian network. A Dynamic Bayesian network is basically a Bayesian network [15] in which some links (called "temporal links") represent time elapsing. Many real world domains need to take into account time elapsing. For some variables the probability distribution on their states is not constant in time. It varies due to the only fact that time elapses. In real world time elapses in a continuous way, whereas in a Dynamic Bayesian network it elapses in a discrete way: as a sequence of time-slices. Temporal links allow to represent the effect of time elapsing between two time-slices. In CHEERUP the general model of the Dynamic Bayesian network has been instantiated in the flowing way. Each session takes place in a respective time-slice. In each time-slice the event E is represented by a variable (node). The variables E, present in the respective time-slices, are connected by temporal links. Given two time-slices: t1 and t2, (t2 > t1) and the event E, it can be stated that P(E=occurred) is greater in t2 than in t1, for the only fact that a time = t2 – t1 has elapsed. The conditions in which a subject passes the time interval between t1 and t2, are represented by selecting the appropriate context states in t2. If in the real world the event E occurs in the time interval between t1 and t2, in the model the event E occurs in t2, and its occurrence is carried out by selecting, in t2, the state "E=occurred". In t2 are also selected the context states representing the conditions in which the subject has spent the first part of the time interval before the E occurrence. Obviously the model involves a reality approximation, approximation that is as smaller as temporally nearer sessions are.

B. Acquiring subject data

A monitoring session (for short, session) starts by asking the domain expert (i.e. the portal user) to enter, for each single context, the related appropriate state that reflects the conditions under which the subject has passed the time since the last session. After the user has provided, for each context, the right state, he/she has to select if, for the current subject, the event E has occurred or not (for the subject). Let us suppose that the user enters *First Cardiac Infarct* = not-occurred. After the whole context-states-acquisition-page has been filled in, the environment performs correctness and coherence checks of the entered data and, if it is all right, inserts them into the database, causing this way the end of the first step and the beginning of the second step. However, if the current session outcome is E=occurred, or the number of cases so far examined are less than the threshold value needed to make probabilistic inferences, then after the first step the last one (i.e. the session termination step) is directly performed. Let us now pass to consider the second step.

C. Inferring probabilistic predictions

This step has to do with the ultimate purpose of CHEERUP, that is: knowing how much probable the occurrence of E will be in the future for the current subject, supposed that the combination of context states defined for the subject at the present time (i.e. the time of the current session) persists even in the future. The Using environment begins this step by displaying a form page for acquiring the combination of context states (initial combination, or "start situation"). In such form page the combination of context states defined in step one is automatically maintained so to provide a start situation (for the simulation) reflecting, by default, the status-quo. This is just a "default" start situation, in fact the user can define a different initial combination of context states (incoherent modifications are not allowed by the environment) having this way at his/her disposal a powerful tool for exploring the consequences in the future of an initial combination of context states different from the default one. Such a possibility of considering "what would happen if" starting from different initial combinations may be a useful support to decision making about the right measure to take for the subject. Once the initial combination is defined the environment passes to build the simulation plan.

C.1 Automatic building of simulation plan

For each simulated future session of the subject it is supposed that each context has the same state it has in the start situation. In case of time-sensitive states the expression "having the same state" has to be intended in semantic terms. As a consequence, the environment properly modifies the temporal parts of time-sensitive states. For example, turning back to the *First Cardiac Infarct* example, let us suppose that the current session for the subject being monitored is the second one (subject age = 62), so that there are still four simulated future sessions before getting to the end of the monitoring process. Let us also suppose that in the first step (subject acquisition data) of the second session the user has entered for the context *obesity* the time-sensible state = *yes-important and such state has lasted for* 22 *years*. For the simulated future session 3 the environment creates the time-sensible state = *yes-important and such state has lasted for* 24 *years*, and so forth as far as the last session. In general, starting from the start situation defined in the current session, the environment, for all the simulated future sessions, creates context states combinations according to the rule: states that are not time-sensible are kept constant, states that are time-sensible are kept "semantically constant", that is their temporal part is automatically properly updated. After that, the environment checks, for each combination assigned to a simulated future session, if the number of cases so far examined (in that future session) is greater than the threshold value, so that probabilistic inferences can be made. The check starts from the first future session and stops as soon as a subsequent future session with a number of cases below the threshold is encountered. The environment, after the simulation plan building, proceeds to calculate and propagate probabilities concerning the simulated future sessions. Let us focus attention on the mathematical model underlying the prediction algorithm.

C.2 Mathematical model underlying predictions

In CHEERUP the mathematical model of prediction consists in a Dynamic Bayesian network that is dynamically built (according to the simulation plan) and executed (when the portal user asks the environment to produce predictions). Let us examine the basic structure of such a network. The contexts defined by the portal builder: C1, C2, …, CN, and the event E are the nodes of the network. Let us use symbols like E_i and $C1_i$, $C2_i$, …, CN_i to denote the event E and the contexts C1, C2, …, CN related to a session i. Let us use the symbol "→" to represent a causal link, so that "A → B" means "A causes B". Let session i and session m be the first future session and the last future session of the simulation plan respectively (of course $i \geq 2$). The Bayesian network used to produce predictions is the one shown in Fig. 1.

For short, let us use "y" and "n" to denote "occurred" and "not-occurred" respectively. Let us notice that E_{i-1}=n (in fact in case it were E_{i-1}=y the Using environment would communicate to the user that there is no future session to simulate). Let us consider a session k. During the session each context has been instantiated to one of its states, that is each context has been assigned one of its states. For short let us represent an instantiation of contexts related to a session k by simply writing $C1_k,\ldots CN_k$. What the Using environment learns (the topic will be examined in sub-section E) at the end of a session k is:

if k > 1,

$$P(E_k = y \mid E_{k-1} = n, C1_k,\ldots,CN_k) \quad (1)$$

whereas if k =1,

$$P(E_k = y \mid C1_k,\ldots,CN_k) \quad 2)$$

Let us assume that for k > 1, if E_{k-1}=y, then E_k=y independently of the combination of context states (in fact, if in a certain time-slice t1 we enter the piece of information "E has occurred", in a subsequent time-slice that piece of information cannot change). E occurrence causes the end of the monitoring process. For example, if the subject has the first infarct in the time period between session 1 (the first session) and session 2, at the time of session 2 the portal user sets *First Cardiac Infarct*$_2$=y. In such a case the monitoring process would stop at the end of session 2.

After the network structure let us pass to examine what network predictions consist in. Probabilistic predictions consist in calculating the probability of E_k=y for each session k of the simulation plan. Formally, let $i \leq k \leq m$, where $i \geq 2$. The prediction for a session k is the value of

$$P(E_k = y \mid E_{i-1} = n, C1_i,\ldots CN_i,\ldots,C1_k,\ldots,CN_k).$$

For k=i such a value is learned by the environment, according to the (1), in fact it is certain that E_{i-1}=n. By using the probability theory let us face the problem concerning the case of $i < k \leq m$. For short let us use the symbol A to denote the sequence:

$$E_{i-1} = n, C1_i,\ldots,CN_i,\ldots,C1_k,\ldots,CN_k$$

By applying the product rule it can be stated that

$$P(E_k = y \mid E_{k-1} = n, A) \cdot P(E_{k-1} = n \mid A) = P(E_k = y, E_{k-1} = n \mid A)$$

and similarly

$$P(E_k = y \mid E_{k-1} = y, A) \cdot P(E_{k-1} = y \mid A) = P(E_k = y, E_{k-1} = y \mid A)$$

Since the two events (E_k=y, E_{k-1}=n) and (E_k=y, E_{k-1}=y) are mutually exclusive, on the basis of the addition axiom we can state that

$$P(E_k = y, E_{k-1} = n \mid A) + P(E_k = y, E_{k-1} = y \mid A) = P((E_k = y, E_{k-1} = n \mid A) OR (E_k = y, E_{k-1} = y \mid A))$$

Since the set of states {E_{k-1}=n , E_{k-1}=y} is exhaustive, we can state that:

$$P((E_k = y, E_{k-1} = n \mid A) OR (E_k = y, E_{k-1} = y \mid A)) = P(E_k = y \mid A)$$

In conclusion we can state that:

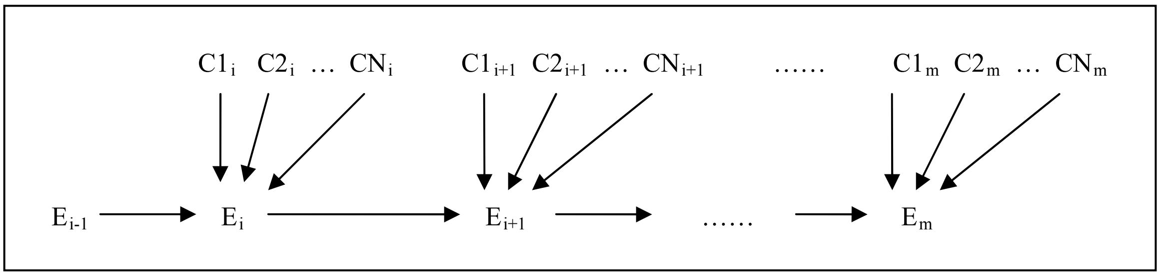

Figure 1. The structure of the Bayesian network used to produce predictions with a simulation plan in which session i is the first future session ($i \geq 2$) and session m is the last one. The arrows connecting E nodes are temporal links.

$$P(E_k = y \mid A) =$$
$$P(E_k = y \mid E_{k-1} = n, A) \cdot P(E_{k-1} = n \mid A) +$$
$$P(E_k = y \mid E_{k-1} = y, A) \cdot P(E_{k-1} = y \mid A)$$

On the basis of these considerations let us state (for short the sequence $C1_i$,…CN_i,…,$C1_k$,…CN_k is represented by $C1_i$,…CN_k):

$$P(E_k = y \mid E_{i-1} = n, C1_i, ..., CN_k) =$$
$$P(E_k = y \mid E_{k-1} = n, E_{i-1} = n, C1_i, ..., CN_k) \cdot \quad (3)$$
$$P(E_{k-1} = n \mid E_{i-1} = n, C1_i, ..., CN_k) + \quad (4)$$
$$P(E_k = y \mid E_{k-1} = y, E_{i-1} = n, C1_i, ..., CN_k) \cdot \quad (5)$$
$$P(E_{k-1} = y \mid E_{i-1} = n, C1_i, ..., CN_k) \quad (6)$$

Let us consider the (3). Every causal path connecting the nodes E_{i-1},$C1_i$,…CN_i,…,$C1_{k-1}$,…CN_{k-1} to the node E_k is a serial structure in which E_{k-1} is the last but one node. Since E_{k-1} is instantiated to a state, each of its antecedents (i.e. the nodes E_{i-1},…CN_{k-1}) does not affect E_k so they can be neglected (according to the Bayesian network theory) and therefore the (3) is equivalent to the (1). The ultimate consequence is that the value of (3) is known: it has been learned by the environment. The value of the (4) is complementary to the value of the (6). The value of the (5) is 1 (as above pointed out). Finally let us consider the (6). The nodes E_{k-1},$C1_k$,…CN_k are all direct causes of the node E_k (we have a converging structure). Since E_k is not instantiated to any of its states, its causes are all independent. Therefore the nodes $C1_k$,…CN_k does not affect E_{k-1} , and as a consequence they can be neglected (according to the Bayesian network theory). The ultimate consequence is that the (6) is equivalent to

$$P(E_{k-1} = y \mid E_{i-1} = n, C1_i, ..., CN_{k-1}) \quad (7)$$

The value of the (7) is the prediction calculated for session k-1. As a consequence the environment produces predictions by performing a probability propagation from session i to session m.

C.3 A numerical example

In order to make abstract considerations easier to understand, let us again use the *First Cardiac Infarct* (for short, FCI) example and let us suppose that session 2 is the present session and sessions 3, 4, 5 are the future sessions of the simulation plan. Moreover, for the sake of simplicity, let us consider two contexts only: *genetic predisposition* (for short, GP) and *obesity* (for short, OB). Finally let us suppose that the initial context states combination is: GP_2 = *no*; OB_2 = *yes-important and such state has lasted for* 22 *years*. So the combinations related to the future sessions will be: for sessions 3, GP_3 = *no*; OB_3 = *yes…*24, for sessions 4, GP_4 = *no*; OB_4 = *yes…*26, for session 5, GP_5 = *no*; OB_5 = *yes…*28. Taking into account that FCI_2=n, predictions (i.e. the probability values of FCI_3=y, FCI_4=y, FCI_5=y) come from executing the Dynamic Bayesian network of Fig. 2.

Let us suppose that the environment has learned:

P(FCI_3=y | FCI_2=n, GP_3 = *no*, OB_3 = *yes…*24) = 0.2
P(FCI $_4$=y | FCI $_3$=n, GP_4 = *no*, OB_4 = *yes…*26) = 0.3
P(FCI $_5$=y | FCI $_4$=n, GP_5 = *no*, OB_5 = *yes…*28) = 0.4

The occurrence probability of FCI=y for session 3 is equal to what the environment has learned (i.e. 0.2). For session 4 it is 0.3 * 0.8 + 1 * 0.2 = 0.44, whereas for session 5 it is 0.4 * 0.56 + 1 * 0.44 = 0.664.

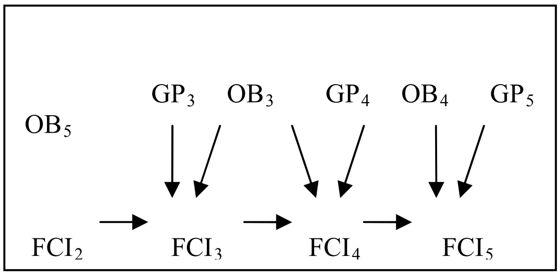

Figure 2. The Dynamic Bayesian network used in the *First Cardiac Infarct* example in order to produce predictions for the future sessions 3, 4 and 5, starting from the present session 2.

D. Presentation of probabilistic predictions

Let us examine the third step of a monitoring session. The environment shows probabilistic predictions in both quantitative and qualitative way. For each future session the related E occurrence probability is shown along with the related qualitative judgment. Moreover, the qualitative presentation is also carried out through a histogram pointing out the trend of future probabilities. The user is also provided by the environment with the possibility of getting answer to the questions: what simulation plan has been built by the environment? Where do the displayed numbers come from? Such predictions represent a power decision-support in order to take suitable measures in advance, measures personalized to the subject under consideration. Moreover, the possibility to compare different predictions resulting from alternative initial states-combinations may help decision making in trade-off problems. For example, let us suppose that E is an undesired event, and that at the current session, for the subject under examination, both contexts (risk factors) C1 and C2 turn out to be present. Let us also suppose that there are two measures: M1 that is effective to eliminate the presence of C1, whereas M2, on the contrary, is effective to eliminate the presence of C2. Let us suppose that the prediction obtained from a start situation in which C1 is absent shows a decrease of E occurrence probability in future sessions that is greater than the decrease obtained form a start situation in which C2 is absent. As a consequence, M1 turns out to be better than M2. Though, if M1 is more expensive that M2, a typical trade-off problem occurs, and probabilistic predictions starting from different simulated start situations can help solve it.

Once the user has collected sufficient prediction information for the current subject he/she proceeds to the termination phase (fourth step) of the current monitoring session.

E. Session termination and learning

Before closing a monitoring session the expert is invited to note some considerations about the subject situation (i.e. measures that should be taken, etc.). The environment shows, for the last time, all the acquired data and asks the user either to definitely confirm or to cancel the whole session. If confirmation is given, the session ends and the learning process is activated.

Let k be the current session and let $C1_k,\ldots CN_k$ be the contexts instantiation (for short X_k) of the session k. Learning consists in updating the value of $P(E_1=y \mid X_1)$ if k=1, the value of $P(E_k=y \mid E_{k-1}=n, X_k)$ if k >1. Such value does not refer to any specific subject, it is general. Learning is accomplished by bringing up to date the quotient

$$\frac{N_{ky}}{N_{ktot}}$$

where $N_{k,y}$ denotes the number of cases that in session k have been found with

$E_k=y \mid X_k$ if k = 1,

$E_k=y \mid E_{k-1}=n, X_k$ if k > 1;

and where N_{ktot} denotes the total number of cases so far examined in session k, given

X_k if k = 1,

$E_{k-1}=n, X_k$ if k > 1.

So far we have examined the environments for building and using predictive monitoring portals. However the environments devoted to these tasks need suitable administering infrastructures to get them to work at best. Let us therefore pass to examine the administering environments.

IV. The Administering Environments

CHHERUP contains three administering environments: the environment for administering portals, the environment for administering subjects, the environment for administering environments.

The Portals Administering environment concerns portals that are in the Using environment already. The environment provides the portal administrator with several utility functions. It is the portal administrator that, among the subjects present in the subjects database, assigns the portal the ones to be monitored. Among the portal administering functions there is the one that allows the administrator to eliminate a subject by the list of the subjects monitored by the portal. This function also re-establishes the statistical situation in the database as if the eliminated subject had never entered the portal.

The Subjects Administering environment contains functions to manage the subjects database and to operate on subjects independently from the specific portals to which they are assigned by portal administrators. Each subject may be monitored by more than a single portal.

The Environments Administering environment is used by the Super-administrator only . The Super-administrator plays the role of general supervisor of CHEERUP. Among the numerous functions of this environment let us notice the group of functions for administering the users of CHEERUP, for example the function to create a new user of CHEERUP. In this case the term “user” has not the meaning of “final user” (i.e. a subject to be monitored). Its intended meaning is that of a person that can use the environments of CHEERUP playing one or more roles: portal builder, portal user, portal administrator, subjects administrator. Another group of functions is constituted by the authorization functions. Once a user of CHEERUP has been created, he/she must be authorized to use a certain environment. An authorization consists in providing him/her with specific userid and password for each environment that he/she has required to use. In other words, the Super-administrator assigns specific roles to the users of CHEERUP. Among the role assignment rules let us notice the following ones. A single subject can be monitored by n different portals (for example, in medicine a subject can be monitored with respect to various undesired pathologic events, and there is a portal for each undesired event respectively), a user can use n portals (e.g. an organization can study the occurrence probability of n undesired events), a single portal can be used by m users (e.g. different organizations, possibly international, can co-operate in monitoring a large subjects population relatively to an undesired/desired event). These features contribute to make CHEERUP a product suitable to be successfully used even in large co-operation contexts, facilitating and structuring co-operation among working groups.

V. Related Work And Discussion

Industry is a typical world in which predictive monitoring, mostly intended as preventive monitoring, has find numerous applications with a variety of approaches. Twenty years ago already, preventive monitoring was a crucial theme for manufacturing processes (typically, for example, in the world of the large car manufacturing companies [1]). In manufacturing industries there is a considerable attention to reduce costly and unexpected breakdowns. As a consequence preventive maintenance is becoming more and more important. Maintenance should abandon the traditional “fail and fix” approach to pass to the more modern “predict and prevent” one [2]. As a consequence the fundamental need is monitoring degradation instead of detecting faults. A predictive performance and degradation monitoring is what is needed for an effective proactive maintenance to prevent machines from breakdown. The theme of degradation monitoring for failure prevention applied to vehicle electronics and sensor systems is faced in [3] where the authors propose a unified monitoring and prognostics approach that prevents failures by analyzing degradation features, driven by physics-of-failure. The need, for manufacturers of complex systems, to optimize equipment performance and reduce costs and unscheduled downtime, gives rise to system health monitoring. System states monitoring is augmented with prediction of future system health states and predictive diagnosis of possible future failure states [4]. Predictive monitoring has been also applied to flexible manufacturing systems. In [5], the main objective is to manage progressive failures in order to avoid breakdown state for the flexible manufacturing system. The approach to predictive monitoring proposed in [6] uses predictions from a dynamic model to predict whether process variables will violate an emergency limit in the future (predictions are based on a Kalman filter and disturbance estimation). Predictive monitoring has also been applied in many specific industry worlds like, for example, press manufacturers [7] and chemical plants [8]. In many industrial applications predictive monitoring assumes the meaning of preventive monitoring and aims to enhance the effectiveness of preventive maintenance by making it proactive. In some cases though, predictive monitoring is finalized to early intervening to maintain a system at a high level of performance. It is the case of a predicting monitoring application for wireless sensor networks: "...by monitoring and subsequently predicting trends on network load or sensor nodes energy levels, the wireless sensor network can proactively initiate self-reconfiguration…" [9]. In most industry applications the acquisition of monitoring data is carried out through sensors [10].

Predictive monitoring has found many applications in medicine too. In general they are specific applications. For example, interesting applications have been carried out in the field of diabetes therapy. In [11] and [12], continuous glucose monitoring devices provide data that are processed by mathematical forecasting models to predict future glucose levels in order to prevent hypo-/hyperglycemic events. Many other specific applications of preventive monitoring may be found in medicine [13], [14]. In [14] the authors present predictive monitoring applied to patients exposed to gentamicin ototoxicity: the most common single known cause of bilateral vestibulopathy. Patients undergoing exercise rehabilitation therapy were tested repeatedly during follow-up visits to monitor changes in their vestibulo-ocular reflex. Predictive monitoring turned out to be useful for continuing or modifying the course of vestibular rehabilitation therapy.

The proposal presented in the paper has the ultimate purpose that is in common with all the cited applications, but, at the same time, has many aspects that distinguish it from them. CHEERUP, is neither a predictive monitoring application nor a general prognostics tool for preventing undesired events in some fields like, for example, manufacturing industries, medicine, etc. CHEERUP does not address a specific domain, it is a general environment for building and using specific applications of predictive monitoring in several heterogeneous domains. Since its approach to prediction is probabilistic, it needs a significant number of cases in order to make statistics sufficiently significant. Moreover it is not an embedded system, it is a tool that has been designed for a use on behalf of domain experts.

VI. Conclusion

Two are the factors that mainly give power to CHEERUP : generality and simplicity. Its generality is due not only to the great heterogeneity of the portals that it is possible to build and use, but also to the possibility of using predictive monitoring as a means not only to prevent undesired events, but also to favour desired events. Its simplicity is due to: friendly user-interface (simplicity in using), modular structure (simplicity inside), correctness and coherence controls (simplicity in assistance), mathematical model underlying prediction (simplicity in theory), explanation of the reason why predictions are what they are (simplicity in comprehension). Finally, let us conclude by mentioning the CHEERUP structural propensity to favor co-operation among working groups by means of several facilities useful to work in team, in structured organizations.

In order to provide a paper reader with the possibility of getting a deeper comprehension of the proposal, CHEERUP is equipped with a self-demo facility, an infrastructure that allows the reader to build and test his/her own demo-portal without interfering with real portals present in CHEERUP. The reader, by clicking on "self-Demo" in the CHEERUP homepage, is provided with an operative guide to implement the medical example (First Cardiac Infarct) presented throughout the paper. The interested reader can find CHEERUP at the Web-address: www.cheerup.it

References

[1] S. Spiewak and M. Szafarczyk, "A Predictive Monitoring and Diagnosis System for Manufacturing", CIRP Annals - Manufacturing Technology, vol. 40, no. 1, pp. 400-403, 1991.

[2] J. Lee, J. Ni, D. Djurdjanovic, H. Qiu and H. Liao, "Intelligent prognostics tools and e-maintenance", Computers in Industry, vol.57, no. 6, pp. 476-489, 2006. [Online]. Available: http://dx.doi.org/10.1016/j.compind.2006.02.014

[3] H. Liao and J. Lee, "Predictive Monitoring and Failure Prevention of Vehicle Electronic Components and Sensor Systems", SAE 2006 World Congress & Exhibition, April 2006, Detroit, MI, USA, Session: Automobile Electronics and Systems Reliability (Part 1 of 2).

[4] R. Kothamasu, S. H. Huang and William H. VerDuin, "System health monitoring and prognostics—a review of current paradigms and practices", The International Journal of Advanced Manufacturing Technology, vol. 28, no. 9-10, pp. 1012-1024, 2006. [Online]. Available: http://dx.doi.org/10.1007/s00170-004-2131-6

[5] F. Ly, A. K. A. Toguyeni and E. Craye, "Indirect predictive monitoring in flexible manufacturing systems", Robotics and Computer-Integrated Manufacturing, vol. 16, no. 5, pp. 321-338, 2000. [Online]. Available: http://dx.doi.org/10.1016/S0736-5845(00)00015-6

[6] B. C. Juricek, D. E. Seborg and W. E. Larimore, "Predictive monitoring for abnormal situation management", Journal of Process Control, vol. 11, no. 2, pp. 111-128, 2001. [Online]. Available: http://dx.doi.org/10.1016/S0959-1524(00)00043-3

[7] S. A. Spiewak, R. Duggirala and K. Barnett, "Predictive Monitoring and Control of the Cold Extrusion Process", CIRP Annals - Manufacturing Technology, vol. 49, no. 1, pp. 383-386, 2000.

[8] J Jeng, C Li, H Huang, "Dynamic Processes Monitoring Using Predictive PCA", Journal of the Chinese Institute of Engineers, vol. 29, no. 2, pp. 311-318, 2006.

[9] A. Ali, A. Khelil, F. K. Shaikh and N. Suri, "MPM: Map based Predictive Monitoring for Wireless Sensor Networks", Autonomic Computing and Communications Systems, Third Int. ICST Conf. Autonomics 2009, Limassol, Cyprus, September 9-11, 2009.

[10] S. C. Choi and P. A. Pepple, "Monitoring Clinical Trials Based on Predictive Probability of Significance", Biometrics, vol. 45, no. 1, pp. 317-323, 1989. [Online]. Available: http://www.jstor.org/stable/2532056

[11] J. Reifman, S. Rajaraman, A. Gribok and W. K. Ward, "Predictive Monitoring for Improved Management of Glucose Levels", J Diabetes Science Technology, vol. 1, no. 4, pp. 478-486, 2007.

[12] C. 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, no. 1, pp. 81-88, 2010. [Online]. Available: http://dx.doi.org/10.1089/dia.2009.0076

[13] J. Chen, T.-Y. Hsu, C.-C. Chen, and Y.-C. Cheng, "Online Predictive Monitoring Using Dynamic Imaging of Furnaces with the Combinational Method of Multiway Principal Component Analysis and Hidden Markov Model", Industrial & Engineering Chemistry Research, vol. 50, no 5, pp. 2946-2958, 2011. [Online]. Available: http://pubs.acs.org/doi/abs/10.1021/ie100671j

[14] D. P. O'leary, L. L. Davis and S. Li, "Predictive Monitoring of High-frequency Vestibulo-ocular Reflex Rehabilitation Following Gentamicin Ototoxicity", Acta Oto-Laryngologica, vol. 115, no. S520, pp. 202-204, 1995.

[15] F. V. Jensen, *An Introduction to Bayesian networks*, London: UCL Press, 1996.