ITC
irst

CENTRO PER LA RICERCA
SCIENTIFICA E TECNOLOGICA

38050 Povo (Trento), Italy
Tel.: +39 0461 314312
Fax: +39 0461 302040
e−mail: prdoc@itc.it − url: http://www.itc.it

FEEDING DATA MINING

Avesani P., Olivetti E., Susi A.

July 2002

Technical Report # 0207−01

Feeding Data Mining

Paolo Avesani, Emanuele Olivetti, and Angelo Susi

ITC-IRST,
Via Sommarive 18 - Loc. Pantè, I-38050 Povo, Trento, Italy
{*avesani,olivetti,susi*}*@irst.itc.it*

Abstract. Data mining is a complex process that aims to derive an accurate predictive model starting from a collection of data. Traditional approaches assume that data are given in advance and their quality, size and structure are independent parameters. In this paper we argue that an extended vision of data mining should include the step of data acquisition as part of the overall process. Moreover the static view should be replaced by an evolving perspective that conceives the data mining as an iterative process where data acquisition and data analysis repeatedly follow each other.

A decision support tool based on data mining will have to be extended accordingly. Decision making will be concerned not only with a predictive purpose but also with a policy for a next data acquisition step. A successful data acquisition strategy will have to take into account both future model accuracy and the cost associated to the acquisition of each feature. To find a trade off between these two components is an open issue. A framework to focus this new challenging problem is proposed.

1 Introduction

Very often there are initiatives to provide inductive evidence as explanation of a complex phenomena although a collection of data is not available in advance. It is straightforward that in this context a data acquisition plan becomes a strategic preliminary or intermediate goal.

To arrange a data acquisition plan could be not trivial if the collection and the recording of information can not take advantage of electronic devices to automate such a process. Moreover the assumption that the effort spent to collect a vector of data is feature independent could be no more sustainable. For example in the agriculture domain a biological test to fill a feature that describes the presence of a particular pest could be really expensive.

The objective of a data acquisition plan is twofold: to increase the opportunity of a much more accurate model for the next step of data analysis and at the same time to lower the costs associated to a data acquisition plan.

It is to be remarked that in this work we assume that the space of features to be collected can change step by step.

This work aims to define a framework for this kind of challenge as preliminary step towards the development of working solutions. Therefore this paper doesn't provide yet a solution to arrange successful data acquisition policies.

Let's start with the next section focusing our attention to the agriculture domain. It will explain the motivations of this work illustrating two research projects in the area of pest control management. Starting from this working scenario the third section will introduce an intuitive definition of the objectives that arise from the previous motivations. A more formal statement of the framework will be provided in the fifth section after a preliminary introduction of the basic definitions. A particular attention will be devoted to the specification of the evaluation process.

2 Working Scenario

The motivations of this work arise from the agricultural domain in which we are involved by two ongoing projects regarding apples production in Trentino (northern Italy): PICO and SMAP projects. Main target of both is pest control to improve production in Integrated Production systems and Organic Farming (low environmental impact); the actors of each project are of the same kind: *researchers* set up models to focus the attention of *technicians* on particular period and location for treatments and to guide them to collect data; *farmers* are advised by technicians to act specific treatments and give feedback to them to improve models.

- PICO[1] (Protezione Integrata Colture Ortofrutticole, Integrated Production of Fruit and Vegetables) project is devoted to the development of a decision support system for apple pest management (Codling Moth). It supports entomologists while developing insects pest models, given a set of biological and meteo data collected on different orchards; it also supports agronomists and producers in tuning the resulting models to the specific environmental characteristics of the local territory. Two main targets, still open issues, are how to let the model follow the changes that occur in the insects population behavior during time, and how to help the experts during the process of selecting the most relevant characteristics that influence the life of the insects that may change over time. It's to notice that results of the model will guide actions that will change the system, leading to the need of a new model, specific to the new conditions. In this way the model will need an updating process due to its interaction with the environment. It is interesting to notice also that the first target can produce a sort of iterative process: starting from the set of environmental and biological data, it produces a model. The use of the retrieved model can influence the environment and the behavior of the disease and can modify them, modifying also the working hypothesis used to generate the previous model, forcing a new step of the process to adapt the model to the new environment.
- Apple Proliferation (AP) plant disease has been a remarkable spread in most of the fruit growing area in Trentino (northern Italy) and south-west Germany in the last 5 years. Apple growers are concerned about the spreading of the infection (due to a phytoplasma) because it leads to complete economic loss of production. At present no curative treatments are known for AP disease. The main goal of the SMAP[2]

[1] This project is supported by the Italian Ministry of Scientific and Technological Research.

[2] The project is supported by Autonomous Province of Trento, Italy.

(Scopazzi del Melo - Apple Proliferation) project is to study AP mechanism, symptom regression and improve tree resistance to disease. Our contribution to SMAP project aims to discover relevant dependencies among collected data between trees, orchards and environment (meteorological, AP carriers presence, closeness to other specific cultivation or wood, geographical characterization etc.) to support and improve data gathering methodology and AP comprehension; a main target of our work, that is an open issue in our research, is to give advices on the way to collect data for future monitoring (which features and the amount of instances to collect), because every source of relevant information is not already known and possible new relevant sources may come out in future due to progresses in the knowledge on AP or changes in the local environment. Data collection is also a critical factor because it is costly (it's a lengthy work done by many actors): to design an acquisition campaign each year is critical for the project and our contribution is devoted to support the decision process to minimize collections costs and improve models accuracy.

The two projects should provide real-life application support to our research and feed the necessary domain knowledge and data to work out experiments.
PICO and SMAP projects involves Data Mining problems because of the amount of data they will manage (increasing in time), and because of the need to obtain reliable models and deeper understanding of the problem, analyzing relations between collected features using statistical approach as well as machine learning techniques.

3 Data Mining and Decision Making

Data mining and decision making are two tasks closely related. Usually from the application point of view their relationship is clear: one is consequential of the other.

Let's consider the agriculture scenario where two entities do exist: an operator, a researcher or a technician, and the environment, the apple plantation. The operator, through the interaction or the observations, collects the evidence of the pest behavior. The collected data will be further exploited to elaborate accurate model of the pest. The ultimate goal is to achieve a predictive capability of the pest behavior.

Taking this perspective the sequence of actions is clear: first the data mining task, that allows to build a model, then the decision support task, that will exploit such a model. The purpose of the inductive process is to support a deliberative process that should take advantage of a better knowledge of the pest.

Let summarize a sketch of the data mining process carried on very often.

1. collecting and preprocessing of the data;
2. data analysis and synthesis of a data model;
3. validation and deployment of the learned model.

The process above is usually accomplished under many tacit assumptions. Let's try to get them explicit step by step.

First of all the data collection is assumed to be not cost-sensitive: it is not taken into account whether it is expensive or time consuming to acquire more data. From this assumption immediately follows an other: the default data acquisition policy is to do oversampling. If any doubt applies concerning with the potential usefulness of a given feature its acquisition is promoted. The favorite policy is to overestimate the collection of features that have to be acquired because the filtering of the unuseful ones is postponed to the following data analysis step . The assumption that the set of features to be acquired could be arbitrarily large is coupled with a similar assumption concerning with the size of the data collection. The size of the acquired data that have to support the inductive step is supposed to be meaningful and representative even after the preprocessing and cleaning phases.

A related assumption underlies the data analysis step. A quite common approach is to manage the inductive process with a strategy that aims to detect and to discard the unuseful features. It is the reason why the data mining is accomplished like a strictly forward sequence of steps where the data analysis is not in charge to address the previous or further step of data acquisition. Although the active learning approach aims to cover this opportunity, the resulting acquisition policies don't affect the data dimensionality, i.e. the features to be acquired. A deeper discussion related to this aspect is postponed to the section 7.

To consider the set of the features fixed it is equivalent to do a close world assumption. Every kind of inductive process can not assume any further features that don't belong to the acquired data. It is consistent with the hypothesis that the data analysis step doesn't have the opportunity to influence the data acquisition. The possibility to collect data respect with new features on demand is not allowed.

Let's try to explain how these tacit assumptions affect the design of an architecture that combines data mining and decision support in an application environment as depicted in the previous section through the PICO and the SMAP projects.

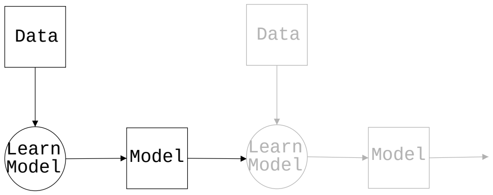

Fig. 1. The current model of data mining.

When researchers and technicians have to face with an unknown apple disease the goal is to build an accurate model of the pest in order to arrange an effective treatment strategy. The first question they have to deal with is the following: what kind of data to collect that potentially could capture the pest model? Decision making requires a support since the early stage and in some way precedes the data mining step. To design an effective data acquisition plan is important for many reasons.

Usually in the agriculture domain it is not possible to acquire large amount of data taking advantage of electronic devices; more often the evidences, like the leaves shape or color, has to be assessed on the field manually. Adopting an overestimate strategy that tends to enlarge the total amount of features to be collected, proportionally increases the risk of noisy data. Under these circumstances it is easy to believe that such a kind of data acquisition could require a strong effort. In this context it's an open challenge to address, since the beginning, the features that seem more promising in playing a role in the inductive process.

An additional drawback affects the data acquisition design. To collect a feature value could be not equally expensive. For example certain feature values can be acquired only through biological tests that could be highly costly. It is clear that in this case a decision support could be helpful to plan whether and when to perform such a kind of test.

Moreover the data acquisition campaign could be crucial for an other reason: it could have additional temporal constraints. For example some data could be collected only in a certain period of the year according to the apple season. The consequence is that if we forget to include a meaningful feature in the current data acquisition campaign we have to wait for the next year.

4 Feeding Data Mining

Data mining is traditionally related to information technology not only because the computational effort required to discover meaningful associations but mainly due to the large amount of data that the electronic devices collect through an automated acquisition. When data mining is applied to fields like the fraud detection or the web log, the data that have to be collected are in some way determined by the software: every event that is traced by an application program will be recorded. When the environment we are going to understand is not monitored by a software a new goal needs to be achieved: to setup a plan that allows to design a model of the data that have to be collected.

A new step takes place when decision making is combined with data mining in the agriculture domain: the design of an incoming data acquisition campaign. Data are no more a precondition but becomes an intermediate target of the whole discovery process. Not only the discovery of an accurate model of the environment behavior but also the discovery of the data design that better enables this achievement.

The revised architecture, as depicted in the figure 2, extends the previous schema introducing a "feed policy". Feeding the data mining becomes an open issue as important as to obtain an accurate model: what kind of input structure provided to the data analysis could allow to enhance the quality of the output?

Let's try to detail better the role of this new step through a sketch of a feed policy that we will have to implement:

1. Select a set of candidate new features to be acquired.
2. Preliminary sampling of the new candidate features.
3. Filling of the most promising predictive features.

The first step is to filter among the huge amount of every possible feature a subset of candidates that will be evaluated through a preliminary assessment. The second point is to rank the subset of candidates doing a kind of subsampling: the goal is to achieve a preliminary estimate of a feature without to accomplish a complete acquisition of the data sample. The third and ultimate point is to complete the acquisition of the most promising features.

Before to discuss the open issues related to this new challenge we have to remark a further revision of the proposed architecture. Up to now another tacit assumption still holds: the data mining is conceived as a “one shot” process. It is not supposed to iterate the basic steps simply because the final result doesn’t affect the data acquisition. At the same time a static view of the world brings to neglect the side effect that a deliberative actor has on the environment behavior using a predictive model.

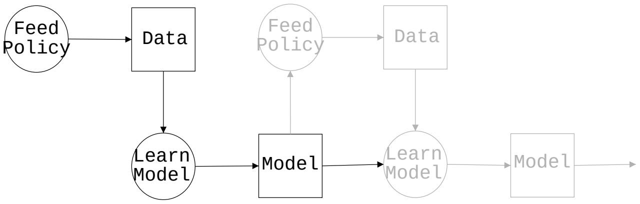

Fig. 2. The “on going” model of data mining.

The real world behaves differently. Usually the predictive models are highly inaccurate but they may start to help a decision making activity. A pragmatic approach suggests to exploit the preliminary result and at the same time to activate an other loop with new steps of data collection and data analysis. After a first loop it is clear that a feed policy can take advantage of a previous collection of data and of the predictive model build on them.

The activity of the researchers and of the technicians in the agriculture domain could be conceived as a yearly process that at each season aims to obtain a predictive model much more accurate of the season before. This result is achieved optimizing the ratio between the costs of the data acquisition campaigns and their ability to detect the most predictive features.

Let’s formalize in detail this new perspective on data mining.

5 Basic definitions

In the following we give some basic definitions to formalize the framework for the “on-going” data mining that will be introduced in the next section.

Let X be the set (possibly infinite) of cases and x_i an instance in the set. Let $f_j : X \rightarrow \mathbb{V}_j$ be a feature, where $\mathbb{V}_j$ is the set of all possible values of the instances over feature f_j. We call F the set of all possible features f_j (possibly infinite).

Let $M : 2^X \times 2^F \rightarrow 2^{\mathbb{V}}$ a function that, given a set of features and a set of instances, gives back a set of values in $\mathbb{V} = \{\bigcup_j \mathbb{V}_j\}$. We call $\mathbb{M}$ the set of all possible functions M. We can represent M as a matrix with possible unknown entries where rows are instances and columns are features.

Given a set of instances $\overline{X}$ and a set of controlled feature $\overline{F}$, $\overline{M}$ is the actual values of the instances over those features.

We define $F^* \subseteq F \setminus \overline{F}$ as the set of new possible interesting features suggested by the domain expert and $\widetilde{F} \subseteq F^* \cup \overline{F}$ the subset of features that as to be selected from $F^* \cup \overline{F}$ to improve model and to guide data collection at the next step. Let be $X^S \subseteq X$ a subsampling set over X that we use to decide whether a new feature $f_j \in F^*$ is good to improve the model or not. Let $\widetilde{X} \subseteq X$ be the set of new instances we add after determining $\widetilde{F}$ and c_{f_j} the cost payed for the introduction of a new feature $f_j \in \widetilde{F}$.

Given this framework we introduce the concept of hypothesis of the evolution of a model and in particular we define $H_k^{\overline{M}}$ as a partial hypothesis at step k of the model evolution over some given dataset $\overline{M}$, $\mathcal{H}$ as the final hypothesis, combination, in a specified way, of the all the $H_k^{\overline{M}}$. The space of all possible hypothesis $H_k^{\overline{M}}$ is said $\mathbb{H}$.

We introduce also the concept of error $\varepsilon : \mathbb{H} \times \mathbb{M} \rightarrow \mathcal{R}$ as an error function of an hypothesis over a given set of features and instances.

Finally we introduce π_f, as the policy to promote a feature from the set of new features F^* (suggested by the domain expert) to the set of selected features $\widetilde{F}$ to be added to the current features under control $\overline{F}$, and π_i, as the policy to promote an instance from the set X to the set $\widetilde{X}$ of the instances to add at the next step.

At every time step of the process k, $\overline{X}_k$, $\overline{F}_k$, $\overline{M}_k$, $\widetilde{X}_k$, $\widetilde{F}_k$ and F_k^* are equivalent definitions as above.

6 A framework for "feeding" Data Mining

Let's organize the basic notions above in a framework to better focus the key problem and to provide an evaluation setting accordingly. First we will introduce the incremental mining process then the related incremental acquisition strategies. The definition of the evaluation criteria and of the related experimental method will conclude the picture.

6.1 Incremental Mining Process

The process to improve the model of the system is iterative and based on successive change of focus on domain, to progressively specialize the model on difficult instances. We start from an initial set of instances $\overline{X}_0$, features $\overline{F}_0$ and their values $\overline{M}_0$, to work out an hypothesis $H_0^{\overline{M}_0}$ and its error rate $\varepsilon(H_0^{\overline{M}_0}, T)$, namely the usual accuracy measure. Note that the accuracy error is estimated over a test set, different from the training set which is used to build H_0; in this sense $\overline{M}_0$ should be considered divided in two parts: $\overline{M}_0 = \overline{M}_0^{training} \cup \overline{M}_0^{test}$ and obviously $\overline{M}_0^{training} \cap \overline{M}_0^{test} = \emptyset$, giving $\varepsilon(H_0^{\overline{M}_0^{training}}, \overline{M}_0^{test})$.

¿From the current candidate set $\overline{F}_0$ and set of new feature F_0^* suggested by the domain expert we get the most promising subset of it ($\widetilde{F}_0$) applying an active feature policy π_f evaluated over a subsampling $X_0^S \subseteq X$. Given this new set of features we add new instances $\widetilde{X}_0$, to the current ones under observations, using an active instance policy π_i to find instances in difficult region of the domain for the hypothesis $H_0^{\overline{M}_0}$. We call $\phi_0 \subseteq \mathbb{V}$ these regions. With $\widetilde{X}_0$ and $\widetilde{F}_0$ it is possible to work out $H_0^{\widetilde{M}_0}$ that should perform better than $H_0^{\overline{M}_0}$ on ϕ_0. With the new sets of features and instances we obtain

$\overline{X}_1 = \overline{X}_0 \cup \widetilde{X}_0$, $\overline{F}_1 = \overline{F}_0 \cup \widetilde{F}_0$ and $\overline{M}_1 = \overline{M}_0 \cup \widetilde{M}_0$; with these new sets and the two hypothesis we can obtain the hypothesis for this new step as a composition of them : $H_1^{\overline{M}_1} = H_0^{\overline{M}_0} \oplus H_0^{\widetilde{M}_0}$, where $\oplus$ means for example that

$$H_1^{\overline{M}_1} = \begin{cases} H_0^{\widetilde{M}_0} & \text{in } \phi_0 \\ H_0^{\overline{M}_0} & \text{elsewhere} \end{cases}$$

In the same way as we got $H_1^{\overline{M}_1}$ we can obtain $H_2^{\overline{M}_2}$ and so on, till a certain $H_k^{\overline{M}_k}$. In the general case we can write the recursive rule:

$$H_{k+1}^{\overline{M}_{k+1}} = \begin{cases} H_k^{\widetilde{M}_k} & \text{in } \phi_k \\ H_k^{\overline{M}_k} & \text{elsewhere} \end{cases}$$

and

$$\begin{aligned} \overline{M}_{k+1} &= \overline{M}_k \cup \widetilde{M}_k \\ \overline{F}_{k+1} &= \overline{F}_k \cup \widetilde{F}_k \\ \overline{X}_{k+1} &= \overline{X}_k \cup \widetilde{X}_k \end{aligned}$$

The goal of active feature policy π_f is to select features from a suggested set F_k^* trying to maximize the estimated hypothesis accuracy built on a subsampling of X constrained to the zone where the previous hypothesis fail; this policy takes into account different costs to collect each feature and try to trade off estimated accuracy gain with budget limitations; this policy focus only on feature costs and can be improved specifying better cost-sensitive mechanism. Now we can give the formal definition:

Definition 1 (Active Feature Policy). *¿From a subset of the feature space F (actually a subset of $F^* \cup \overline{F}$), a dataset $M \in \mathbb{M}$ and and an hypothesis H^M the active feature policy π_f determines the subset of features that could be useful for the improvement of the model at the next mining step:*

$$\pi_f : 2^F \times \mathbb{M} \times \mathbb{H} \rightarrow 2^F$$

The goal of active instance policy π_i is to select instances from regions of whole set X with poor accuracy performance of current hypothesis, getting feature constraints on the data acquisition step.

Definition 2 (Active Instance Policy). *¿From a particular instance in X, a dataset $M \in \mathbb{M}$ and an hypothesis H^M the active instance policy π_i determines if the instance is useful for the improvement of the model at the next mining step:*

$$\pi_i : X \times \mathbb{M} \times \mathbb{H} \rightarrow \{0, 1\}$$

6.2 Incremental Acquisition Strategies

We can have different learning processes depending on the way we implement the policy of active feature selection π_f or active instance selection π_i:

- if we stress only π_f and do not acquire any other instances we have $\widetilde{F}_i \cap \overline{F}_i = \emptyset$ (and $\overline{X}_0 = \overline{X}_1 = ... = \overline{X}_k = \overline{X}$) and call it *Backward* strategy. In this case we favour the detection of new promising features constraining the data acquisition to the previously stored instances extending their description.
- if we stress on π_i and restrict π_f to $\widetilde{F}_i \subseteq \overline{F}_i$, in every next improvement step we add only new instances, taking their values only on a subset of the current features (some useless feature can be dropped here and taken back in a future step). We call this case *Forward* strategy, because we have a new dataset each step.

In the general case no restriction are assumed on the policies and the process has a first step of feature selection based on an estimated evaluation worked out on a subsampling dataset that is acquired to focus on regions where the current hypothesis fails some instance.

6.3 Evaluation Criteria

Given this framework the model can be evaluated and exploited giving it an instance as a set of features defined by the last $\overline{F}_k$. The model perform the recursive composition of the hypothesis at each step when evaluates $H_k^{\overline{M}_k}$.

The sequence of features in input, used to perform the evaluation, can be in the same order they were introduced building the model; if we want to minimize the cost of the query we have to build a *feature tree* based on the single feature costs and the degree of coverage of the instance distribution by each $H_i^{\overline{M}_i}$ that compose the last hypothesis $H_k^{\overline{M}_k}$ $(i < k)$.

A general rule to evaluate the decision taken to do the single improvement step has to take into account the results both of the policies and of hypothesis induction; in this sense we state that the relation to be satisfied is

$$\varepsilon(\overline{H}_i^{\overline{M}_i}, T) \leq \varepsilon(\overline{H}_i^{\widehat{M}_i}, T)$$

where $\widehat{M}_i$ is the full matrix of all values of every feature in $\overline{F}_i$ on every instance $\overline{X}_i$; the T set is a test set (not biased) independent from the improvement process. This means that in this framework we try to induce a model without having access to all data, still reaching good accuracy, even better than the full-information case; the key point is that useless information can slow the learning process, and selecting only informative instances and features can speed-up the process, leading to a better accuracy in fewer steps.

6.4 Cost Mining

Let's focus on the role of cost associated to data mining; in this framework we are interested only on the cost related to the data acquisition step, that is the leading cost in our working scenario. Two factors are related to the costs of data acquisition: the number and type of features that describe an instance and the number of instances that has to be collected. We introduce the notion of cost associated to a given feature c_{f_j}, that is the cost of getting feature f_j for a certain instance (we assume that each instance has the same cost respect to a given feature); so the active feature selection policy π_f shall trade off the accuracy of the model and the cost needed to acquire the data to produce it. We estimate the savings of a policy by the ratio

$$\frac{|\overline{M}_i|}{|\widehat{M_i}|} \cdot \frac{\varepsilon(\overline{H}_i^{\overline{M}_i}, T)}{\varepsilon(\overline{H}_i^{\widehat{M_i}}, T)}$$

where $|M| = \sum_{j=1}^{N}(c_{f_j}|m_j|)$ is the cost of dataset M and $|m_j|$ is the number of entries of column m_j of M; here we relate the cost of the datasets used by the iterative process ($\overline{M}_i$) to the cost of the full matrix ($\widehat{M_i}$, the full-information case) and the accuracy of the hypotheses built on them.

6.5 Evaluation Process

To evaluate the model produced and to compare different implementations of the decision policies, we can think about two alternative strategies : *on line* versus *off line*. They can be non mutual exclusive and can be referred as *run-time* or *incremental* evaluation and *a posteriori* evaluation.

The on line approach assumes that $\widehat{M_i}$ is not available but only $\overline{M}_i$. Due to the fact that the decision process determines at each step which is the best direction (features and instances) to take and there isn't any possibility to step back and change previous decisions; but avoiding to get back and try different alternatives doesn't allow to compare other policies, so other solutions seems to be taken.

The off line approach assumes a full availability of $\widehat{M_i}$. This is the case of pre-existing datasets to use for performance testing purpose. $\widehat{M_i}$ is first partitioned in two datasets (train and test), and the simulation of adaptive iterative process is performed on the train one. An initial set of instances and features is extracted and then a fixed number of sampling and filling steps are performed on available data exploiting the full information accessibility. We propose two possible alternatives about the target to reach in the off line approach to evaluate strategies:

- We fix the number of the mining steps and compare different solutions (policies) giving them homogeneous samplings: a fixed amount of instances is acquired at each step to maintain equally representative samples needed for future comparisons. Another assumption we need to be able to compare is to use the same modeling technique (induction trees, neural networks etc.) The target optimization to be compared between different solutions is the costs/accuracy of the resulting models. The different stress over costs or accuracy can modulate the target to satisfy application needs.

- We fix the amount of budget that every solution can spend for the learning phase, so the number of instances that can be taken at each step is constrained by the costs of features selected by the feature policy. After spending all the budget, maybe in a different number of steps, alternative solutions can be compared respect to the target of accuracy that should be maximized. With this method, different approaches that prefer feature oriented acquisitions (more features, less instances), can be fairly compared with approaches that prefer extensive samples with few features.

7 Related work

Two main research areas dealing with our issue are *Feature Selection* and *Active Learning*. Feature Selection operates to look for the most relevant subset of features to focus the attention while modeling the system ([4],[6]). Active Learning looks for relevant examples (instances) to be labelled, during the learning process, discriminating them respect to useless examples, which cannot give new information to improve current hypotheses ([3], [11], [10]). Active Learning and Feature Selection reduce the number of instances and features for the learning phase and this is useful for 4 main reasons:

- learning is computational intensive, so reducing data speeds up the process and implies more efficiency.
- labelling cost is high, so asking the expert (teacher or oracle) is not always possible due to limited resources.
- as the learning process requires time, we can reach better accuracy more quickly selecting better examples and features ([2]).
- data collection is expensive even without labelling, as we have said in our working scenario: reducing the cost of data collection can be a main target in real life applications.

In [2] every Feature Selection strategy is classified with respect to the policy of adding/removing features, the organization of search in feature space, the policy to compare alternative features to get the more useful ones and the stopping criterion of the search. Our approach proposes a forward selection, because it starts with few features and tries to add new ones, with the possibility to remove some useless old features at each step; the organization of the search in feature space is guided by the domain expert and it is an open point where to propose new solutions; comparison between possible sets of features is performed by a *wrapper* method ([6], [4]) based on a subsampling of instance space and an induction algorithm to work hypothesis to estimate possible gain in accuracy; no stopping criterion is given because the process is meant to follow new needs of improved accuracy in time or changing of the environment.

As in Active Learning community, we specify which instances are useful to focus the attention on to cover problematic region of the domain, but we haven't full control over the instance space ([3]), so we can give only *feature constraints* to advice data collector. The way to translate feature constraints in specific instances in the domain is still an open problem that remains outside this framework nowadays.

Another work on dynamical aspects of data sampling focus the attention on the fairness of the database sample [5] and assume that the database exists and is accessible;

we relax this hypothesis proposing another important aspect of data: the collection cost as the main limitation on sampling.

Because of the presence of costs in inductive learning we face to a broad literature on the subject ([12], [13]), but we restrict to cost of cases to acquire and policy to take care of it (as in [8]), shifting to a different direction of cost-sensitive learning, concerning instances and features, not well stressed until now.

8 Conclusions

In this paper we have presented the motivations that, moving from the real experience on agriculture domain, detect a new demand for extended decision support related to data mining. We focused the problem of data acquisition plan as a relevant problem where the environmental setting doesn't provide the opportunity to automate the collection of data. A framework to formally define the notion of data acquisition policy is proposed.

The next step will be concerned with the design of a working solution. Alternative hypotheses will be evaluated following the method illustrated in the last section. The datasets that we are processing in the PICO and SMAP projects will enable a realistic experimentation of the relationship between model accuracy and acquisition costs. Moreover a direct comparison will be carried on between the innovative feature-based policy and the instance-based policies available in literature.

References

1. A. Blum. Relevant examples and relevant features: Thoughts from computational learning theory, 1994.
2. Avrim Blum and Pat Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97(1-2):245–271, 1997.
3. David A. Cohn, Les Atlas, and Richard E. Ladner. Improving generalization with active learning. *Machine Learning*, 15(2):201–221, 1994.
4. George H. John, Ron Kohavi, and Karl Pfleger. Irrelevant features and the subset selection problem. In *International Conference on Machine Learning*, pages 121–129, 1994. Journal version in AIJ, available at http://citeseer.nj.nec.com/13663.html.
5. George H. John and Pat Langley. Static versus dynamic sampling for data mining. In Evangelos Simoudis, Jiawei Han, and Usama M. Fayyad, editors, *Proc. 2nd Int. Conf. Knowledge Discovery and Data Mining, KDD*, pages 367–370. AAAI Press, 2–4 1996.
6. Ron Kohavi and George H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324, 1997.
7. P. Langley. Selection of relevant features in machine learning. AAAI Fall Symposium on Relevance, pp. 140–144., 1994.
8. T. RayChaudhuri and L. Hamey. An algorithm for active data collection for learning – feasibility study with neural network models, 1995.
9. M. Saar-Tsechansky and F. Provost. Active Sampling for Class Probability Estimation and Ranking. *Machine Learning*, 2002. (to appear in).
10. Maytal Saar-Tsechansky and Foster J. Provost. Active learning for class probability estimation and ranking. In *IJCAI*, pages 911–920, 2001.
11. H. S. Seung, Manfred Opper, and Haim Sompolinsky. Query by committee. In *Computational Learing Theory*, pages 287–294, 1992.

12. Peter Turney. Types of cost in inductive concept learning. In *Proc. of Workshop on Cost-Sensitive Learning at ICML2000*, pages 15–21, 2000.
13. Peter D. Turney. Cost-sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm. *Journal of Artificial Intelligence Research*, 2:369–409, 1995.