Predictive Classification with Imbalanced Enterprise Data[·]

Sophia Daskalaki, Ioannis Kopanas and Nikolaos M. Avouris
University of Patras, Greece

Abstract

Enterprise data present several difficulties when used in data mining projects. Apart from being heterogeneous, noisy and disparate, sometimes they are also characterized by major imbalances between events of different classes. Predictive classification using imbalanced enterprise data necessitates methodologies, which are adequate for such data, especially for training algorithms and for evaluating the resulting classifiers. It is therefore important to experiment with several class distributions in the training sets and a variety of performance measures, which are known to expose better the strengths and weaknesses of classification algorithms. In addition, combining classifiers into schemes, which are suitable for the specific business domain, may very well improve predictions. However, the final evaluation of the classifiers must always be based on the impact of the classification results to the enterprise which can take the form of a cost model that reflects requirements of the enterprise and existing knowledge. In this chapter, taking as example a telecommunications company, we provide the methodological framework for handling enterprise data during the initial phases of the project, as well as for generating and evaluating predictive classifiers. Moreover, we provide the design of a decision support system, which embodies the previously described process with the daily routine of a telecommunications company that struggles to prevent customer insolvency without risking customer relations.

[·] Early draft of a chapter to appear in Liao, T.W. and E. Triantaphyllou, (Eds.), Recent Advances in Data Mining of Enterprise Data: Algorithms and Applications, World Scientific, Singapore, 2008.
To be cited as:
S. Daskalaki, I. Kopanas, N.M. Avouris, Predictive Classification with Imbalanced Enterprise Data, in Liao, T.W. and E. Triantaphyllou, (Eds.), Recent Advances in Data Mining of Enterprise Data: Algorithms and Applications, World Scientific, Singapore, pp. 147-187, 2008.

1.0 Introduction

Continuous advances in digital information capturing, processing and storage technology have altered considerably common practices in most business environments. The current trend for digitization of information using technologies like scanners, sensors, image recognition techniques, etc. as well as ubiquitous computer support and automation of enterprise activities have resulted in the collection of tremendous amount of data on a daily basis, which can be stored and processed in high speeds. This data is potentially a valuable source of knowledge and an important asset for the companies concerned. For this reason, companies often attempt to exploit the data they own with the ultimate goal to gain an advantage in the competitive market they belong.

Data Mining and Knowledge Discovery tools and techniques have been developed during the last years in order to facilitate the underlined process. However, key questions are associated with such an objective. How easy is to extract and operationalize the knowledge that is hiding in the data? Can we expect that the process of extracting knowledge from the enterprise data will be fully automated in the near future? What form may the extracted knowledge take? Will it be simply a Study Report identifying trends and patterns in the data and making recommendations for future actions? Alternatively, can it take the form of a Decision Support System that will aid humans in daily decision making process? When is the decision of building such a system going to be made during the knowledge discovery process and on what criteria can be based?

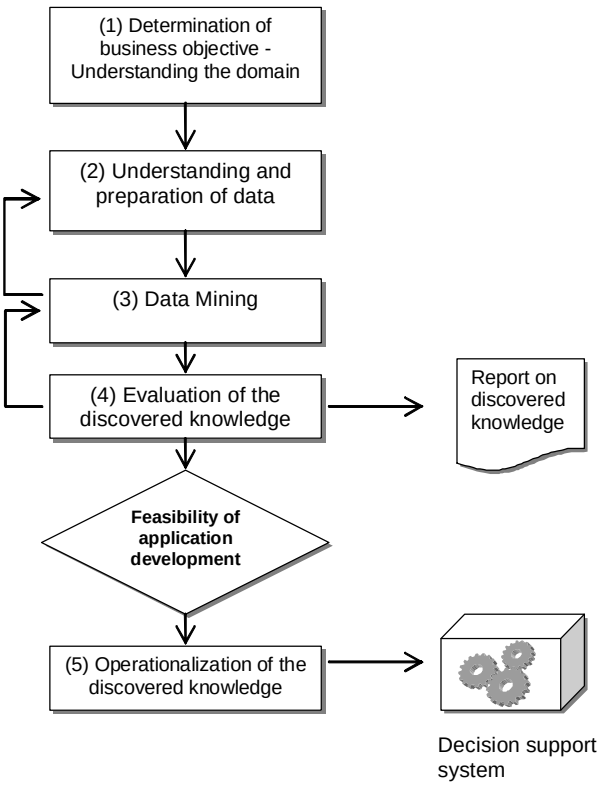

Fig. 1 Abstract model of the Knowledge Discovery from Data Process

In order to build a Decision Support System that is based to a great extend on knowledge extracted from large amounts of data, it is certain that one has to go through a typical Knowledge Discovery from Data (KDD) process. KDD is defined as the "nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data" (Frawley et al., 1991). In other words KDD is the multi-step process, that involves understanding the domain, preparing the data, identifying the function to be applied, choosing

the right algorithms, searching for patterns, evaluating and interpreting the revealed knowledge and finally incorporating it into the decision making process (Fayyad et al., 1996). There have been many attempts to describe this process beyond Fayyad's original nine step model (e.g. CRISP-DM, Cabena et al. 1998, Cios and Kurgan 2005), while Brachman and Anand (1996) discuss the process from an analyst's perspective.

The common aspects of all these models, shown in figure 1, are: the phase of understanding the domain, where the business objectives are determined; collection, preparation and analysis of the enterprise data; data mining, which refers to the application of algorithms for the extraction of patterns[1]; the evaluation of the discovered knowledge and finally the use of knowledge. Fayyad et al. divides further Phase 2 of figure 1, where most of the effort of the KDD research is put, as well as Phase 3, which is in the heart of the process. Instead, little attention has been put in Phase 4 and particularly on the decision to proceed or not with the operationalization of the acquired patterns. However in business environments, it is this phase that will probably yield results worth the effort of the whole KDD exercise and permit re-use of the acquired knowledge beyond the simple description of identified patterns. It is often argued that it is this latter case that any knowledge is obtained. For instance, Wilson (2002) has commented, that knowledge is a term highly misused and often taking the meaning of derived facts (data) or information, when these facts are embedded in a context of relevance to the recipient, while knowledge is directly related to the mental process of comprehension, understanding and learning. So, one may argue that only when the discovered patterns are inserted in the business process and take the form of operational tools that can be applied in every day situations, real knowledge has been extracted from the original data.

In this chapter, we review the process of Data Mining and Knowledge Discovery in large enterprise datasets and focus on the phase of evaluating the discovered knowledge in order to decide whether it is worth proceeding with the development of a Decision Support System based on the discovered knowledge. It is our goal to demonstrate the complexity of this phase in connection to the typical decisions that need to be made. Our proposal is based on previous experience and lessons learned from a data mining and knowledge discovery project that had the objective to manage customer insolvency in the telecommunications industry (Daskalaki et al., 2003). We further provide an example of the form and functionality of a Decision Support System that can be built if a decision is been made that it is worth proceeding with operationalization of the discovered patterns.

2.0 Enterprise Data and Predictive Classification

Telecommunications companies are typical examples of enterprises that accumulate gigabytes of data and for this matter take advantage of technology advances, which continues to provide faster storage devices with higher capacity and lower prices. However, despite the modern techniques in database management systems and data warehouse systems, like in most real-world problems the data are almost never ready for knowledge extraction and the stages of data cleaning, pre-processing and formatting require a considerable amount of effort. Blind application of data mining methods to unprocessed data may result in patterns that are not valid, leading to improper interpretations and wrong conclusions. It is therefore important that the KDD process, when applied to a certain problem, is characterized by many loops that connect the middle and final steps with earlier ones. The iterative character of the KDD process has been stressed by many researchers (Brachman and Anand, 1996), while it is estimated that only a small portion (15-25%) of the total effort needs to be devoted to the actual application of data mining algorithms (Brachman et al., 1996, Zhang et al., 2003).

Difficulties with real-world data, including incompleteness, noise in data, inconsistencies, etc., cause distortion in the patterns that are hidden in the data, low performance of the algorithms and poor quality for the outputs. Apart from these well

[1]Since the term Data Mining (DM) is often used in the literature to describe the whole KDD process, a proposal has been made to use the term Data Mining and Knowledge Discovery (DMKD) instead, in order to avoid confusion (Cios and Kurgan 2005)

documented problems however, in classification problems it is quite common that data may exhibit additional more subtle problems, like major imbalances between classes, uneven or unknown misclassification costs, rarity of the events of interest, etc. The detection of oil spills from satellite radar images (Kubat et al., 1998), the detection of fraud in mobile communications (Fawcett and Provost, 1997) or in the use of credit cards (Chan et al., 1999), the customer insolvency problem (Daskalaki et al., 2003), the prediction of failures in some manufacturing processes (Riddle et al., 1994) and the diagnosis of rare diseases (Laurikkala, 2001), are problems that exhibit at least one of these problems. In presence of these difficulties, several remedies have been suggested including modification of the data set by undersampling or oversampling methods, considering other than the usual performance measures and of course combining classifiers with techniques like bagging or stacking.

In this section we discuss first the general problems that enterprise data carry and concentrate further on the problems that influence predictive classification.

2.1 Characteristics of enterprise data.

It is well known that real data inherently carry a number of problems, which make the KDD process very laborious. From our experience with the customer insolvency problem in telecommunications and several other classification problems, the following characteristics are known to cause the majority of technical difficulties:

(a) *Noise*: Noise in data is a well documented problem for real-world problems. In a study on the sensitivity of machine learning algorithms in noisy data, Kalapanidas et al. (2003) report that Decision Trees performed better than other classification algorithms. Several techniques have been proposed as a remedy for the problem of noise, including wavelet denoising (Li et al. 2002). However, it is almost certain that noise can not be completely eliminated. For the telecommunications industry for example, it is known that a case of insolvency may be the result of a fraudulent act or due to factors that the customer cannot control (health problems, personal bankruptcy, etc.). The success of predicting insolvency is based on the premise that customers change behavior during a certain period, therefore the second group of customers will add noise to the first group, unless they are distinguished. However, most companies will not carry such information; therefore any classification effort cannot be expected to achieve high scores of accuracy.

(b) *Missing Data*: This is another well known problem that appears extensively in real-world datasets. Missing values in data are related to non applicability of the field, unavailability of the value, data corruption or not timely insertion of the value (the well known problem of the semantics of the "NULL" value in data bases). However, machine learning algorithms cannot tackle datasets with missing values. Therefore, various techniques need to be applied for filling the empty fields. These include replacing the "NULL" value with a default value or with the mean value of the specific characteristic over the whole dataset or the mean value of the specific characteristic over a given class.

(c) *Limited and distorted view of the world.* In any data mining project the available datasets represent the real world entities (for example, the individual customer of the enterprise) in a limited and distorted way. In our case, the source of our information was exclusively the telecommunication company and the information that this company can maintain on its customers. For various ethical and legal reasons, this information cannot be inter-related to other sources of information; so for instance, the customer is represented as user of the particular service, with no means of revealing other social or financial aspects that might had influenced the individual's behavior patterns and might had strong impact towards an insolvent behavior.

(d) *Overwhelming amount of secondary characteristics.* In the usually very large data set involved in a KDD project, often deduced from transactional data or other sources that register interaction of the company with its clients, many parameters may be defined. In a modern telecommunications company this data may very well characterize the behavior of customers. Selection of only relevant parameters for the problem of interest is a tedious process that can be partly based on statistical analysis tools and techniques and partly on

understanding the significance of the parameters in the problem by the analysts involved. Therefore, identifying a subset of these parameters, and subsequently using adequate tools for selecting the most relevant ones is a crucial and important phase, often directly related to the particular problem and not easily reproducible.

2.2 Problems arising with predictive classification

Apart from the problems presented previously, enterprise data often carry additional characteristics that influence adversely predictive classification and turn the data mining stage of the KDD procedures into a much more difficult task than it usually is. The characteristics that make certain problems difficult are the following:

- Uneven distributions for the different classes.
- Small number of cases of insolvent customers (minority class) in the dataset
- Different and often unknown misclassification costs for the two classes.

3.0 The process of Knowledge Discovery from enterprise data

As already mentioned in the Introduction, the process of Knowledge Discovery from Data (KDD) has been modelled in various ways. The 9-step framework (Fayyad et al., 1996), presented in Figure 2, was found most suitable for describing our experience with the customer insolvency problem and is used in this section as a frame for reviewing the process. Since our final goal was to develop a Knowledge-based Decision Support System for managing insolvency, the KDD process was applied to confirm the hypothesis that prediction of customer insolvency is possible by studying patterns in the data provided by the telecommunications company. Although researchers aspire to fully automated processes for all steps involved, the discovery of knowledge from data with little intervention or support of domain experts cannot always be true. In fact in (Brachman and Anand, 1996) it is admitted that the domain knowledge should lead the KDD process.

```
(1)     Learning the application domain
(2)     Creating a target dataset
(3)     Data cleaning and preprocessing
(4)     Data reduction and projection
(5)     Choosing the function of data mining
(6)     Choosing the data mining algorithm(s)
(7)     Data Mining
(8)     Interpretation
(9)     Using discovered knowledge
```

Figure 2. The 9-step framework of the KDD process according to Fayyad et al., 1996

Taking one phase at a time, we briefly describe the actions that need to be taken, along with our experiences from the insolvency prediction problem. Emphasis is given to the role of the domain expert throughout the whole project. From the early stages, it was well understood that a number of domain experts and sources of data had to be involved in the process. Domain experts, e.g. executives involved in tackling the problem of customer insolvency, and salespersons who deal with the problem in a day-to-day basis were interviewed during the problem formulation phase and their views on the problem and its attributes were recorded. An investigation of the available data was also performed and this involved executives of the departments of information systems and corporate databases who could provide an early indication on sources and quality of available data. Other key actors for this project were the data analysts, who were also involved together with the knowledge engineers and data mining experts.

3.1 Definition of Problem and Application Domain

During the initial phase of each project, the specific characteristics of the problem need to be defined and the objectives or goals for the whole project need to be set. The role of domain experts during this phase is evident and very important.

For our project, for example, we had to define the term "insolvency prediction" in a way that made sense to the telecommunications company and this meant that the prediction ought to take place early enough, when there is still time for preventive and possibly aversive measures. The billing process of the company, the rules concerning overdue payments and the currently applied measures against insolvent customers had to be explicitly described by the domain experts. Moreover, the objectives had to be defined with the help of decision makers or domain experts. Setting the objectives in a given project is a very important task of this phase and influences heavily the selection of performance measures later in the data mining phase. As will be apparent further on, the performance measures play a decisive role for the evaluation of classifiers. For our problem, three objectives were set as prevailing for the company:

1. Detection of as many insolvent customers as possible.
2. Minimization of the false alarms, i.e. the number of good customers that are falsely classified as insolvent.
3. Timely warning about possible insolvencies, so that prediction can be useful in business terms.

The first objective is evident, given that the insolvent customers cause loss of revenue for the company; therefore detection of as many as possible of them is of prime importance. However, an even more important objective is to maintain the good relationship that exists with the good customers (second objective). In other words, the company should take actions if and only if a customer is classified as insolvent with high certainty. Otherwise, the company takes the risk of loosing good customers. It turns out that these two objectives are conflicting, thus reducing false alarms causes further reduction in the number of customers predicted to become insolvent after the next due date for payment. As for the third objective, this influenced the data that were collected from the corporate data sources. Our experiment investigated the hypothesis that in the case of customer insolvency, calling habits and phone usage in general change during a critical period before and right after termination of the billing period. Therefore, this objective indicated the need for data that exhibit usage of the service in regular intervals much shorter than the billing periods.

3.2 Creating a target dataset

During this phase, the data that are analyzed throughout the project ought to be determined. It is an important stage because critical decisions must be taken. Decisions concern the type of data that are needed to fulfill the objectives, the timeframe during which data will be collected and the subset of the actual population on which the study will take place. The role of domain and business knowledge in this stage concerns the structure of the available information and the semantic value of it. The key actors for this stage come from the data processing department, mainly employees involved in data entry and processing activities for the relevant information systems.

Customer behaviour, in particular, may be described by numerous characteristics, most of them not readily available. For our research purposes, two kinds of data were requested. The first group referred to detailed customer information (like name, occupation, address, etc.), derived from contract files and phone directory entries. The second group referred to time-dependent customer data providing information on the telephone usage (from the so called CDRs, i.e. Call Detail Records) and on financial transactions (like bills and payments). Unfortunately, no more details of customer credit conditions were available in the corporate databases, neither could become available from outside sources. In order to make this study more representative, a cross-section of customers was used; data were collected from three different geographic areas, one rural, one small town and one of a major urban centre. In terms of time, the data in the target dataset covered a span of 17 months.

As often occurs in KDD projects, the data came from several different sources (databases) of the organization; however, they were all integrated and kept in a suitably designed data warehouse built for this purpose. For purposes of confidentiality and protection of customer privacy, data were codified in a secure way. As an indication for the size of the data warehouse built, collected data were over 10 GB in raw form.

3.3 Data Cleaning and Preprocessing

At this stage, it was essential to test the quality of the collected data, to inter-relate the heterogeneous data items in the data warehouse, and to filter out information of no significance. Data cleaning is a tedious process, however if not performed it is impossible to proceed to the data mining phase, where data are assumed to be of good quality and very relevant. During this phase also the domain knowledge is very important and the role of domain experts very critical.

Since the size of the collected data in such type of project is usually very large, it is helpful to reduce data, if possible. For example, in our study a 50% decrease in data volume was achieved by eliminating all calls that were charged less than 0.3 €. The elimination of this data was not affecting our final goal for insolvency prediction given that the company is mostly interested on detecting patterns of expensive calls placed from certain customers with the ultimate goal not to be paid. The domain expert in any project should also be responsible for such actions and decisions, which are extremely important since they manage to purify the data towards the upcoming application of the data mining algorithms. Furthermore, with the help of domain knowledge, the elimination of irrelevant attributes is also possible, even at this stage, before the actual feature selection that takes place in the data mining phase. For example, in our case, the attribute "amount charged" in each bill was considered irrelevant, since it is known that not only insolvent customers relate to high bills, but also very good solvent customers. Instead, large fluctuations of the amounts charged in consecutive bills were considered as more relevant to insolvency and were taken into consideration.

Additionally in this stage, certain error correction procedures need to be applied, in order to sanitize the data from missing or erroneous values. Such problems are unavoidable with real data and in most cases are due to the dispersion of data sources and lack of consistency among the information systems within the organizations that provide the data. Accordingly, synchronization of data is a very important and tedious procedure during this step. For our project, it was necessary to study the calling habits of all customers during a period starting several weeks before a billing period expires. However, different groups of customers belong to different billing periods, thus forming different *groups of phone accounts*. Therefore, insolvencies appeared in several points of time during the period of our study and in order to study behavioral patterns of insolvent customers putting events in a time scale relative to the end of a billing period was necessary. All data cleaning and preprocessing activities that take place during this phase should be guided by domain experts from all departments involved with the underlined procedures.

3.4 Data Reduction and Projection

The data cleaning and pre-processing tasks during the previous phase expose for the first time the problems that may exist with the collected data. As a result, in the current phase, the data are further reduced, if necessary, and several new features may be added following actions of data transformations as a result of statistical inferences on the primary data. For the transformations or projections of the original data, again the objectives of the project and domain knowledge ought to be the actual guides.

For the customer insolvency problem, data pre-processing revealed a number of cases of insolvent customers with insufficient information that eventually had to be eliminated from the data set. New attributes that measure fluctuations in telephone usage or exhibit overdue payments had to be created through transformations of the original data. The study period of the behaviour of insolvent customers was set to be approximately a period of seven months before the actual disconnection of their phone. The decision on the length of the study period was based on data analyses and business requirements. Within the seven-month study period,

the call transactions made by each customer were aggregated in two-week periods, according to certain aggregation functions (sum, count, average, standard deviation, etc.). This procedure deduced several new attributes that were calculated for all customers, solvent and insolvent.

In addition, with the help of statistical inference at this point, a number of features were tested and those that did not provide any valid or useful information in distinguishing solvent from insolvent customers were eliminated. All tests were performed in the original dataset, however in order to create customer profiles, and even more so in order to study the usage of the phone for customers of the two classes it was important to create a smaller and more manageable dataset projecting all characteristics from the original one. The new dataset included information about 28,220 customers out of which only 196 were the insolvent ones. For each customer the data included: (i) two attributes for the customer's profile (static information), (ii) sixty-six attributes for the usage of the phone over fifteen consecutive two-week periods, and (iii) four attributes for the financial transactions of the customer (payment and agreements for payment with instalments). Therefore, seventy-seven attributes were collected in total for each customer in the dataset.

3.5 Defining the Data Mining Function and Performance Measure

During this phase, the purpose of the knowledge to be derived from data mining is defined and this in turn defines the data mining function, whether it is classification, clustering, regression, association rules, etc. as well as the performance measures to be used for evaluating the function.

In many projects, it is possible that the data mining function can be defined at an earlier stage. In others, however, the analysts may define initially a set of possible functions and it is only after the data preparation and pre-processing tasks that the decision for the function is finalized.

The problem of predicting customer insolvency was viewed as a *two-class classification problem*, where each customer could be classified in one of two classes: *most possibly solvent* or *most possibly insolvent*. As classification problem, it carried some of the characteristics discussed earlier:

1. In the dataset that resulted from the preprocessing stage, approximately 99.3% were negative and 0.7% positive cases. Thus, the distribution between the two classes was very uneven.
2. The absolute number of insolvencies in the data set was very small, because only a few insolvency cases arise in every given billing period.
3. The misclassification costs for the two classes of customers, although unknown, were not the same.

Classification problems with such characteristics are particularly difficult to solve. As suggested in (Weiss and Provost, 2001; Chan and Stolfo, 1998) new datasets have to be created, where the distribution of customers between the two classes is altered. In our case, the new distributions between classes were achieved by undersampling, i.e. maintaining all cases of insolvent customers in the original data set, while performing a stratified sampling for the solvent customers. Our goal was to create a representative sample of the solvent customers, so that the algorithms could be trained sufficiently well. Therefore, the triad of characteristics *geographic area*, *type of phone connection*, and *group of bills* were used as sampling strata. These three characteristics had to be carried in the sample with the same proportions as in the original dataset, in order: (a) to maintain the three distinct geographic areas in the reduced data set, mentioned previously; (b) to represent the different types of phone connections; and (c) to eliminate seasonality associated with billing periods. Using this procedure several datasets were created, each with a different class distribution, which were further used for experimentation with machine-learning algorithms.

As mentioned earlier, the class distribution in the dataset was highly imbalanced (1:142), while this ratio may vary with time and geographic area. As for the misclassification costs, it is known that companies are interested in predicting as many insolvent-to-be

customers as possible, however they may prefer to miss a portion of "bad" customers than to hassle a large number of "good" customers (Ezawa and Norton, 1996; Daskalaki et al., 2003). In technical terms, this means that false alarms (falsely predicting a solvent customer as insolvent) are highly undesirable, because companies do not want to put at risk their relations with good customers. This information, which originated from the decision makers in the company, defined the *business objectives* for our study. Even though it would had been of some help, the objectives were not quantified further, so our effort was to match the business objectives with the performance measures used for the evaluation process.

For classification problems with high imbalance in class distribution as well as for problems with unknown class distribution, the *Average Accuracy Rate* is not an appropriate performance measure (Provost and Fawcett, 1997). This is explained in datasets with two classes, out of which one is rare, because the error that stems from the minority cases is disproportionally larger compared to the error that stems from the majority cases. Thus even accuracy rates close to 100% may not be satisfactory (Weiss and Provost, 2003). Alternative measures and evaluation strategies have been suggested and these include the *ROC analysis* (Provost et al., 1998), the *Area under Curve (AUC)* (Chawla, 2003), and the *Geometric Mean* (Kubat and Matwin, 1997) or the *F-measure* (Lewis and Gale, 1994) of the accuracy rates for the majority and the minority classes. All these measures calculate the accuracy rates for the two classes separately and then attempt to combine them in a way that both rates can play some significant role. An additional performance measure that combines better the two objectives set forth earlier (i.e. high True Positive and low False Positive) is the *Precision rate*. Precision, which gives the percentage of the correctly predicted cases out of the total number of cases named to belong to a given class, measures the ability of a classifier to be more precise with its predictions. In our case one would be interested in maximizing both the precision and true positive rates for the minority class and additionally the true negative, the accuracy rate for the majority class.

3.6 Selection of data mining algorithms

At this stage of the KDD process, the data mining algorithms are selected. For each data mining function, the fields of statistics and machine learning provide many alternative algorithms, which differ quite a lot in terms of their model representation. Selecting the data mining algorithms depends on several factors including the type of data, the analyst´s preferences and competences, and also availability and popularity of certain tools.

In our research effort, we experimented with several alternative classification algorithms for testing our hypothesis. The algorithms used in our study were initially the *Discriminant Analysis*, *Decision Trees* and the *Multilayer Perceptron Neural Networks*, and later during a broader experimentation, we additionally used *Linear Logistic Regression, Multinomial Logistic Regression, Bayesian Networks*, and *Support Vector Machines*. Each one of them gives a different classifier, for example, a linear one is produced by the Discriminant analysis, a non-linear by the neural network, and a rule-based by the decision tree. These algorithms were initially applied to the dataset that carried 196 cases of insolvent and 28,024 of solvent customers (a proportion of 1:142) and the results are shown in Table 2.

Classification Algorithm	True Positive Rate	True Negative Rate	Precision Rate
Discriminant Analysis	0.3077	0.9864	0.1402
Neural Network	0.0204	0.9998	0.4444
Decision Tree	0.0000	0.9999	0.0000
Bayes Network	0.6429	0.9259	0.0572
Multinomial Logistic Regression	0.0153	0.9998	0.3750
Linear Logistic Regression	0.0000	1.0000	N/D
Support Vector Machine	0.0000	1.0000	N/D

Table 2. Classification results when the "natural" distribution is used for training

As one may see, not all algorithms managed to train classifiers when applied to such an extremely imbalanced dataset. In fact, only the Discriminant Analysis (with a 30.77 % for TP) and the Bayes Networks (with a 64.29% for TP) appeared to overcome this problem successfully, while the Multilayer Perceptron Neural network and the Multinomial Logistic Regression exhibited only a very low performance for the minority class (2.04% and 1.53%, respectively). The other three algorithms treated insolvent customers as noise and classified all cases of the test sets in the 10-fold cross validation procedure to the majority class. In order to overcome the problem caused by the "natural" distribution, datasets with the artificial distributions 1:100, 1:50, 1:25, 1;15, 1;10, 1:5 and 1:1 were constructed. For this purpose, the original dataset was split into a testing set (using 25% of the data) and training set (with the remaining 75% of the data). The new datasets were built using the stratified random sampling procedure discussed earlier for constructing the new training sets.

3.7 Experimentation with data mining algorithms

In this stage, the actual searching for patterns of interest takes place with the help of the chosen data mining algorithms. Particularly for predictive classification, the results are summarized in confusion matrices, which provide the true and false positives, and true and false negatives. For our project, several experiments were realized in order to test and compare the performance of the aforementioned classification algorithms using a 10-fold validation process splitting the data in training and testing data, as discussed in section 3.6. First, simple performance measures were used (the accuracy and precision rates) and then the composite performance measures AUC, geometric mean, and F-measure. In this chapter, we provide a summary of our experimentation, while more details can be found in (Daskalaki et al., 2006).

Using the simple measures it is concluded that the accuracy rate for the minority class (*TP*) has the tendency to increase as the proportion of positive examples in the dataset increases, while at the same time the accuracy rate for the majority class (*TN*) and more so the precision rate (*PR*) for the minority decrease. These observations are true roughly for all classification algorithms but the Bayesian Network. This algorithm appears to be insensitive to changes in the class distribution, while it gives the best values for the *TP* and the worst ones for *TN* and *PR*. From these results, we conclude that none of the algorithms prevails in performance based on these three measures. Moreover, it is clear that maximizing the *TP* rate conflicts to the maximization of the *TN* or *PR* rates. This can be justified as follows. It is expected that the higher the percentage of positive cases in the training dataset, the higher the probability of positive prediction from an induced classifier, thus both the number of false positive cases and the number of true positive cases are expected to increase. Moreover, if the percentage of the increase in the true positive cases is higher than the corresponding percentage for the false positive cases, then PR decreases. Apparently, since the majority class outnumbers severely the minority class, even a small increase for the minority is proportionally larger compared to an increase of the same size in the majority class.

The performance of the classification algorithms was also evaluated using three composite performance measures, the *AUC* that uses both *TP* and *TN* (*FP* = 1 – *TN*), the geometric mean of *TP* and *PR* ($\sqrt{TP \cdot PR}$), and the *F*-measure of *TP* and *PR* $\left(F = \frac{(b^2+1) * TP * PR}{b^2 * PR + TP} \right)$.

According to our study, the *AUC* measure behaves very similar to the *TP* rate and increases as the proportion of minority cases in the training set increases. Apparently, the classifiers induced by Bayes Networks gives the highest *AUC* value for all different class distributions. Conversely, the geometric mean and the *F*-measure exhibit an approximately concave behavior for most algorithms and attain maximum values when the class distribution is in the range of 1:25 to 1:5.

For the geometric mean, this behaviour is explained because as we have mentioned earlier, the *TP* rate increases with the number of minority cases in the dataset and the *PR* rate decreases for the same changes. An increase of the geometric mean indicates that the

achieved increase in *TP* is quite beneficial because it is not accompanied by a "large" decrease of *PR*. The geometric mean attains a maximum at that class distribution where the benefit from the increase in *TP* rate is larger than the corresponding decrease in the *PR*. Using the geometric mean as performance measure, the classifiers induced by the Neural Network algorithm exhibit superior behavior for almost all class distributions, by achieving maximum performance at the 1:25 dataset. The rest of the classification algorithms behave in a comparable fashion and attain maximum performance either at the 1:10 or 1:5 dataset. The only exception we observe is due to the classifiers induced by the Bayesian network algorithm, which as already discussed, are insensitive to changes in the class distribution, therefore the geometric mean is approximately the same for all class distributions.

The F-measure (Lewis and Gale, 1994) also combines the rates *TP* and *PR*. It also depends on the β factor, which is a parameter that takes values from 0 to infinity and is used to control the influence of *TP* and *PR* separately. It can be shown that when $\beta = 0$ then *F* reduces to *PR* and conversely when $\beta \to \infty$ then *F* approaches *TP*. Based on the *F*-measure, the classifiers induced by the Neural Network prevail by giving the highest values for several datasets followed by the classifiers from the Decision Tree algorithm and the Linear Logistic Regression. The classifiers from Bayes Network give approximately the same value (close to 0.1) for all class distributions and the classifiers induced by Multiple Logistic Regression give the smallest values for all class distributions except only of the dataset 1:1 where it gives the highest. In addition, the datasets in the range 1:25 to 1:5 appear to train the classifiers in a way that achieve the highest *F* values for all algorithms.

3.8 Combining Classifiers and Interpretation of results

In this stage, the results of the experimentation that took place in the previous stage are interpreted and if necessary, certain previous steps are repeated afresh, while ensemble techniques that involve combination of classifiers are used for improving the performance.

For our study, judging from the whole range of class distributions we examined (from the set with the "natural" to the set with the "balanced" distribution) using six classification algorithms and several performance measures, it is clear that the measures *TP* and *PR*, which better reflected the business objectives of the problem, are conflicting to each other. The geometric mean $\sqrt{TP * PR}$ and *F*-measure of the *TP* and *PR* manage to combine the two measures in an effective way and compare the different classifiers. As suggested by these two measures, the classifiers induced by the Neural Network algorithm may provide better overall predictions for the insolvent customers. They are followed by the Decision Tree algorithm and the Linear Logistic Regression. For further improving the predictive capability of our classifiers, we decided to let the three classification algorithms compare with each other on a case-by-case basis and thus combine their results into voting schemes. The goal for this last effort was to achieve stronger reassurance for our predictions and an improved basis for an operational decision support system that gives safe predictions on possible customer insolvencies.

3.8.1 Voting Schemes of classifiers

From a case-by-case comparison of the results of different classifiers many combination models are possible to be developed. For our study, three such voting schemes were attempted to further improve the predictive capability of the three best classifiers. Rule #1 (R1) is a democratic rule, where any given case is classified to class i, if two or more classifiers vote i. Rule #2 (R2) is a veto rule for the majority class, where any given case is classified to the minority class, if and only if all three classifiers vote for the minority class; otherwise the case is classified to the majority class. Rule #3 (R3) is also a veto rule but for the minority class, i.e any given case is classified to the majority class, if and only if all three classifiers vote for this class; otherwise the case is classified to the minority class. For this study, the three voting schemes were applied in conjunction with the classifiers that were induced by the algorithms of Neural Networks, Decision Trees and Linear Logistic Regression for all different datasets (class distributions 1:142 to 1:1).

A comparison of the performance exhibited by the voting schemes shows that R3 gives the best *TP* rates and *AUC*, while at the same time it gives the worst *TN* and *PR* rates. Exactly the reverse is observed for rule R2, while the democratic rule R1 is always between the other two. This behavior of the rules was expected since R3 is conservative towards the negative class and R2 is conservative towards the positive class. Given that apart from high *TP* rate, in this problem we are also looking for high *PR* rate, it is once more clear that combinations of *TP* and *PR* should be examined for the final decision. The results indicate that collaboration between classification algorithms may prove useful for problems like fraud detection and prediction of customer insolvency, where the cost of misclassification for good customers is high and the decision maker requires high degree of assurance in the advice of the prediction tool, in order to take possible action.

3.9 Using the Discovered Knowledge

In this stage, the newly discovered knowledge is evaluated from an operationalizational point and the main focus is on integrating the new knowledge with the existing domain knowledge. In case the enterprise needs a knowledge-based decision support system to aid the execution of certain business activities, two more issues need to be resolved. The first concerns the cost effectiveness of the classification scheme that will be used for prediction and the second and most important the actual design of the system. The following two sections of this chapter are devoted to these two issues, the evaluation of classifiers using cost criteria and the design of an Intelligent Insolvencies Management System intended for the Telecommunications Industry.

3.9.1 Development of a cost-based evaluation framework

Following the evaluation procedure of the induced classifiers in section 3.8, it is clear that selecting the optimal classifier is not an easy task. Having experimented with several classification algorithms and combinations of them, trained with a wide range of class distributions and calculating many performance measures, it should be observed that the picture remained blurred for the decision makers. In reality, the performance measures used are exclusively based on the contents of confusion matrices that count cases of correct and incorrect predictions. For real world problems that concern enterprises, it is necessary to tie performance measures with the economic impact of such predictions. This can be done by defining a cost/gain matrix and a cost/gain function. Both of them are defined uniquely for a given problem or set of problems and their definition requires the involvement of domain experts. As mentioned earlier, for problems like the fraud detection or the customer insolvency prediction, the objective is to maximize the *TP* and minimize the *FP* and according to this statement, the gain function was defined to be:

Total Gain (*TG*) = Gain from Positives (*GP*) – Loss from Negatives (*LN*)

This function is based on a fact that predicting correctly bad customers is translated into revenue for the company and hassling good customers by falsely taking actions against them may result in loss of money. Let us assume the following two cost measures (Table 3): C_P, the expected gain per customer that is correctly classified as insolvent and C_N, the expected cost per customer incorrectly classified as insolvent, respectively. Then the gain function can be written as follows:

$$TG = a \cdot C_P - c \cdot C_N$$

where *a* and *c* represent the number of true positive and false positive cases, respectively in the confusion matrix. Moreover, when the class distribution of a given dataset changes by throwing away only majority cases, then the value for *c*, the number of false positive cases, must be multiplied by the normalizing factor, where N_0 is the population for the majority class in the original dataset and N_a is the population of the majority class in the dataset with the artificial distribution. Then the gain function takes the format:

$$TG = a \cdot C_P - c\left(\frac{N_0}{N_a}\right) \cdot C_N \tag{1}$$

If instead, the testing set carries the "natural" distribution the normalizing factor $\frac{N_0}{N_a}$ equals to one. Using non-zero costs only for the true positive and the false positive cases is quite a reasonable structure of the cost/gain matrix for this type of problems (Zadrozny and Elkan, 2001). The meaning of C_P and C_N can be expressed through some functions $f_i(w_i, z_i)$, $i = 1,2$, where w_i represent weights with values in [0,1]. These weights express the percentage of the insolvent-to-be customers that are expected to be turned around and eventually recover or a percentage of the solvent customers characterized as insolvent that may be distressed and cause loss to the company by stepping away. Similarly, z_i may represent the average amount in their monthly bills for each class separately. While the cost coefficients are not easy to calculate exactly, as will be shown later in this section, it is only the relative gain C_P / C_N that is necessary for indicating whether predictions provided from a classification scheme will result to profitable solutions for the company.

		Predicted	
		Positive (P)	**Negative (N)**
Actual	**Positive (P)**	C_P	0
	Negative (N)	C_N	0

Table 3. The cost/gain matrix used for the customer insolvency problem

First, we give two conditions that are necessary to hold for a classifier to be of interest.

Proposition.
Given a cost/gain matrix with the structure of Table 3, then any classifier must satisfy the two conditions below

$$\frac{c_n}{a} < \frac{C_P}{C_N}, \quad \text{or equivalently for the rates} \quad \frac{FP}{TP} < \frac{C_P}{C_N} \cdot \frac{p(+)}{p(-)} \tag{2}$$

and

$$PR_n > \frac{1}{\frac{C_P}{C_N} + 1} \tag{3}$$

in order for the gain function (1) to be greater than zero.

For equation (2) $c_n = c \cdot \left(\frac{N_0}{N_a}\right)$ denotes the normalized value for the number of false positive cases in the confusion matrix, while $p(+)$ and $p(-)$ are the prior probabilities for the positive and negative cases, respectively, in the dataset.

Using conditions (2) and (3), the outcome of a classification algorithm can be related directly with the profitability of a system that potentially may be developed using the results of the algorithm. It is suggested that in order for the *TG* to be positive, it is necessary for any candidate classifier to provide a value for $\frac{FP}{TP}$ which is smaller than the relative gain per insolvent customer compared to the loss per misclassified solvent customer times the fraction of the prior probabilities in the dataset. Similarly, the precision rate must be quite high if the

relative gain C_P / C_N is small and conversely, if the fraction C_P / C_N is large, then the precision is allowed to take smaller values.

The impact of the fraction C_P / C_N in the total gain becomes more distinct if we write the gain function as a line equation:

$$\frac{TG}{C_N} = a \cdot \frac{C_P}{C_N} - c_n \qquad (4)$$

Given any classifier it is now easy to plot the line represented by eq. 4 for different values of C_P / C_N. Every classifier may define such a line using the information that is provided from its confusion matrix. In case we need to compare a number of classifiers then we have to track the point where each line crosses the horizontal axis (given by $\frac{c_n}{a}$) and the slope of each line. The first is important because it is from that point on that the total gain takes positive values and the second because it gives the rate of change for the total gain. Thus for our problem even if the relative gain from correct classification is not known, given a large number of classifiers one may still find the set that maximizes the total gain for different values of C_P / C_N.

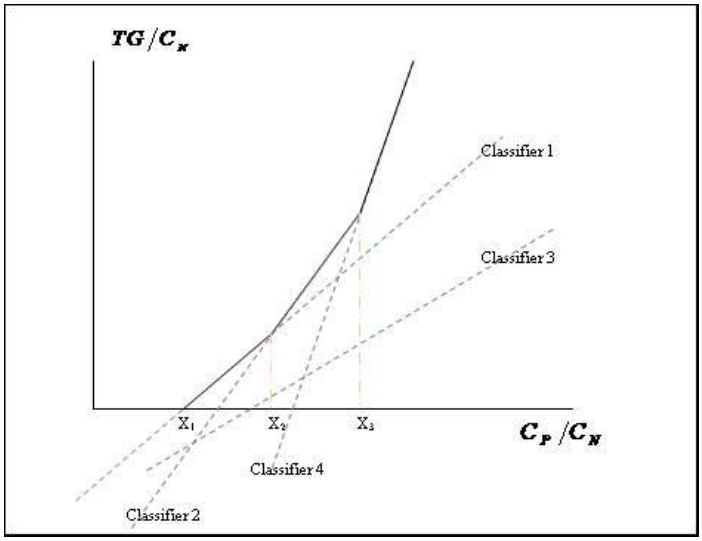

Figure 3. The optimal classifiers for the customer insolvency problem

Let us, consider three hypothetical classifiers, which provide the three lines shown in figure 3. It is clear that there is no classifier that is worth considering if the value for C_P / C_N is less than x_1. However, if the fraction C_P / C_N takes values in the interval [x_1, x_2), then classifier 1 is the best choice, because it crosses first the *x*-axis. For values larger than x_2 classifier 2 is the best choice, because it promises higher values for the gain function. For the same reason, when the C_P / C_N takes values in the interval [x_2, x_3) classifier 1 is the best choice. It is obvious that classifier 3 never prevails; therefore, it is never a good classifier for the underlined problem. This procedure may be repeated until there is no classifier left with higher slope. The result is a multi-segment line where each line segment represents the classifier of best choice for some specific interval in the definition set of C_P / C_N.

This approach was developed in (Daskalaki et al., 2006) to study the economic impact of classification in the customer insolvency problem and bears similarities to the methodology presented in (Drummond and Holte, 2000b; Drummond and Holte, 2004), where the normalized cost curves are plotted against the probability cost function. In addition, it is shown that the cost curves are the duals of certain points in ROC space (Provost and Fawcett, 2001) and the optimal cost curves form the dual representation to the ROC convex hull. The cost curves are more informative, however, because they indicate the range of class distributions and cost fractions where a given classifier dominates over the others.

C_P / C_N	**classification algorithm**	**performance measure**	**class distribution**
2.0 – 2.4	*Voting Rule #2*	*Precision* OR *F* (β = ¼)	1:50
2.4 – 3.3	*Voting Rule #2*	*Precision* OR *F* (β = ¼)	1:25
3.3 – 8.1	*Voting Rule #1*	*G. Mean* OR *F* (β = ½ or β = 1)	1:25
8.1 – 18.1	*Neural Networks*	*G. Mean* OR *F* (β = 2)	1:25
18.1 –	*Voting Rule #3*	*G. Mean* OR *AUC* OR *TP* OR *F* (β = 5)	1:15
39.7	*Voting Rule #3*	*G. Mean* OR *AUC* OR *TP* OR *F* (β = 5)	1:10
39.7 –	*Voting Rule #3*	*AUC* OR *TP* OR *F* (β > 5)	1:5
75.5	*Voting Rule #1*	*AUC* OR *F* (β > 5)	1:1
75.5 – 313	*Voting Rule #3*	*AUC* OR *TP* OR *F* (β > 5)	1:1
313 – 547			
> 547			

Table 4. Best classifiers for different values of the relative gain C_P/C_N

For the customer insolvency problem, this procedure was applied for all previously mentioned classifiers. The result, which is shown in Table 4, gives the set of best classifiers for a large range of values of the relative gain C_P / C_N. Specifically, Table 4 indicates that if the relative gain is less than 2.0, then it is not profitable obtaining prediction of insolvent customers with any of the available classifiers. If the relative gain takes values in the interval [2.0, 2.4] then the classifier induced by R2 and trained in the dataset with the artificial class distribution 1:50 is the best choice. Similarly, in the interval [2.4, 3.3] the classifier induced again by R2 and trained in the dataset 1:25 is the best choice. It is worth noting that in both cases, these classifiers give the highest ***PR*** value in their respective datasets. Proceeding as previously, in the interval [3.3, 8.1] the classifier induced by R1 in the 1:25 dataset gives the best classifier. Going back to the performance measures, the chosen classifier gives the highest geometric mean value in the 1:25 dataset and this performance makes it prevail in the examined interval. For larger values of the fraction C_P / C_N the classifiers induced in turn by the neural network in the dataset 1:25, the voting rule R3 in the 1:15 dataset and later in the 1:10, 1:5 and 1:1 datasets are selected as best. These classifiers provided the best performance for the measures geometric mean, *AUC* and *TP*, respectively.

The conclusion of this process is that different classifiers prevail depending on the value of the relative gain C_P / C_N. These classifiers have been trained using different class distributions in the training dataset and different algorithms or combinations of them. In business terms, our conclusion can be translated as follows: If predicting insolvent customers and taking actions against them is very risky (very small values for the fraction of costs C_P / C_N) it may be wiser not to proceed with any classifications. For a little less risky environments (small values for the fraction of costs C_P / C_N) the classifications should be very precise and in order for this to happen the training dataset should be as close to the "natural" distribution as possible. As the risk in the business environment fades away, classifiers with better performance in measures like the geometric mean, which is influenced both by the *PR* and the *TP*, *AUC* and plain *TP* are recommended. Respectively, the class distributions in the training datasets need to be more balanced in order to achieve higher accuracy in the positive cases.

4.0 Operationalization of the Discovered Knowledge: Design of an Intelligent Insolvencies Management System

In this section, we discuss the final stage of the process, which concerns the operationalization of the discovered knowledge in the form of a System that supports decisions for the Enterprise. A crucial decision to be made at this stage is related with the

feasibility and cost effectiveness of the development of such a system. Technical and economical factors need to be carefully studied for this matter.

One should take in consideration that the discovered knowledge, even in the case that it is of high relevance to the mission of the Enterprise and of high quality in terms of applicability, validity and time invariability is not directly usable. There is a considerable effort that is needed for the knowledge to take the form of a software component that integrates with the rest of the informational infrastructure of the Enterprise and incorporates existing business knowledge of established rules, practices etc, relevant to the specific business process.

In the rest of this section, we outline the characteristics of a Decision Support System (called *Intelligent Insolvency Management System – IIMS*), that intends to operationalize the knowledge discovered through the process discussed in section 3 for our case problem. This process yielded a number of interesting results that are used in the Decision Support System. In summary, the products of the process have been the following:

(a) A data model for the insolvency problem, which contains a number of interesting features derived through transformation of primary data, found in the Corporate Information Systems. The features concern customer characteristics, telecommunication traffic data, billing data etc. During the process, it was found that these features could be used effectively for predicting, with acceptable accuracy and precision, future insolvent customers. However, in order for this data model to be operational for future use, a software component should be built that contains a data base implementing the data model and software components that interface with the Corporate Information Systems. This is called *Intelligent Insolvency Management System Data Base (IIMS DB)*.

(b) A number of classifiers have been developed and tested, which have been trained using historical data with different class distributions and techniques and have known performance when tested against subsets of these historical data. The classifiers may take as input new cases of customers and can predict if they are suspect for insolvency or not. In order for these classifiers to become operational, they need to take the form of a Library of software components, called *Insolvencies Predictors (IP)*. Meta-data concerning their expected performance should be stored in the Data Base of the Decision Support System under design.

(c) A cost model which has been developed, visually expressed in the figure 3, that takes as input the fraction C_P / C_N, where C_P and C_N represent an estimate of the expected gain per customer that is correctly classified as insolvent and the expected cost per customer incorrectly classified as insolvent. Given this ratio, the cost model can decide first if a prediction of insolvency can produce financial benefits and in case of positive answer, to suggest the best classifier to be used from the Library of Insolvencies Predictors. This cost model should take the form of a software component (*Cost Model CM*) that controls the selection of the best classifier, given the cost parameters, and error margin.

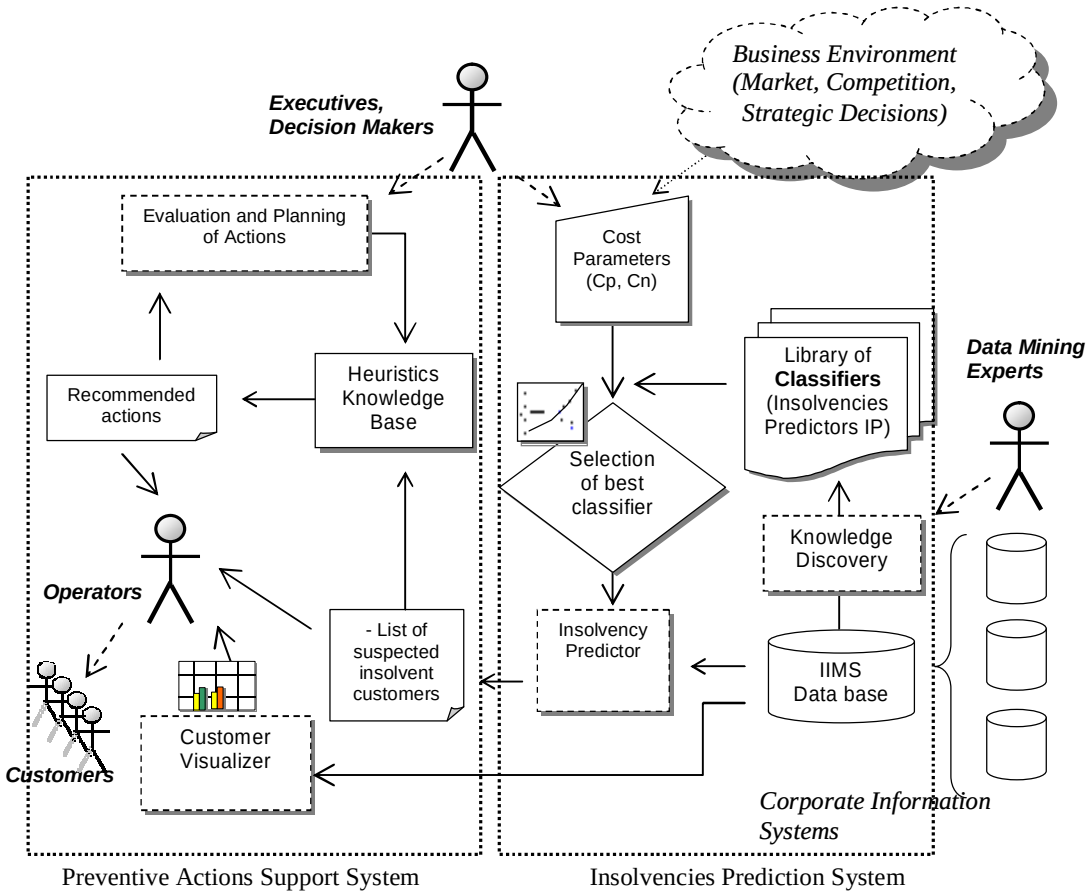

Figure 4. The architecture of the Intelligent Insolvencies Management System (IIMS)

These components, once developed, need to be integrated in the Intelligent Insolvency Management System (IIMS). IIMS involves two distinct subsystems, one related to the Insolvencies Prediction task and one to the Preventive Actions task. A high level architecture of the system is shown in figure 4. The main actors that interact with this system are the *Executives* of the Enterprise who are involved in supervision tasks and setting up the parameters of the Cost Model, a direct consequence of strategic decisions related to risk, competition, market forces etc. The *Operators*, who interact mainly with the Preventive Actions Subsystem, receive recommendations by the Insolvencies Predictor of suspected customers, through the Insolvent Customer Visualizer (ICV) and interact with the Heuristic Knowledge Base (HKB) that contains business rules on preventive actions. Finally the *Data Mining Experts* supervise the Knowledge Discovery Process, which involves receiving feedback on the performance of the classifiers and when needed proceed with retraining the Insolvencies Predictors with more recent data from the Corporate Information Systems. The actors, tasks, components and data flow of IIMS are shown in figure 4.

In a typical use case of the system, the Insolvencies Predictor is triggered by the decision maker who wishes to investigate possible future insolvencies in a set of customers. The best classifier produces a list of suspected insolvent customers. This list is presented to the Operator for inspection, while it is also fed to the Heuristic Knowledge Base which groups the suspected customers in various risk categories, using heuristics derived from background business knowledge of the customer insolvencies department. Specific actions are associated to these risk categories. Before any action takes effect, suggestion for detailed inspection of customers characteristics is made to the decision makers. The Insolvent Customer Visualizer is the component that supports this activity. The actions may involve

warning to the predicted insolvent customers, issue of interim bill, request for a deposit as a guarantee, provisional suspension of the service etc. The Evaluator of Preventive Actions Module records the actions that are effected and their impact to the Company, thus the heuristics are constantly updated as a result of this monitoring by High level Decision Makers.

In Daskalaki et al. (2004) an example of use of the IIMS prototype is discussed. According to this example, the user defines a new heuristic for grouping the suspected insolvent customers and stores it in the Heuristic Knowledge Base. The user through an intuitive interface defines the heuristic rule. An example of such rule is "*Business customers, having made telephone calls to "090" numbers to be considered of high risk for insolvency*". This is a heuristic for defining customers of high risk, for which the company would prefer to take certain preventive action. There is a natural language interface for the Operators to express rules of this nature. This is part of the user interface that permits visualization of user characteristics and definition of rules that permit application of preventive actions to different groups of suspected future insolvent customers.

5.0 Summary and Conclusions

In this chapter, the difficulties of discovering knowledge from enterprise data are reviewed using as an example the problem of predicting customer insolvencies in a telecommunications industry. The main conclusion from the initial stages of the KDD process is that data handling (including collection, preparation, cleaning and preprocessing, reduction and transformation) can be very strenuous and time consuming. However, these first stages are very important in order for the results of the data mining process to be of any value. The role of domain knowledge during this phase is very critical and the presence of domain experts extremely valuable.

Through the data mining stage in several well documented classification problems, it has been found that class imbalance may cause additional challenges in the training of algorithms and in the evaluation of classifiers. Using as an example the customer insolvency problem, it has been argued that the evaluation of classifiers is not possible unless the economic impact of the classification is taken into account. As a result, a set of optimal classifiers can be selected according to the value of the relative gain from correctly classifying positive cases compared to the cost of incorrectly classifying negative cases.

The conclusions from the evaluation of classifiers using the proposed cost model are summarized as follows. First, combining classifiers induced from different algorithms into voting schemes results in classifiers that perform in generally better than single classifiers. Moreover, a veto rule for the majority class has the potential to provide more precise predictions for the minority class (i.e. high precision rate); conversely, a veto rule for the minority or a democratic rule, have the potential to provide more accurate predictions for the minority class (i.e. high true positive rate). Second, in order for these classifiers to achieve their maximum performance they should be trained using datasets with suitable class distributions. The most precise predictions for the minority class are achieved using datasets with class distributions that are closer to the "natural" distribution, while the most accurate predictions for the minority class are achieved using datasets that are closer to the balanced distribution. Lastly, the performance measures that ought to be used for evaluating classifiers must change according to the value of the relative gain C_P / C_N where C_P and C_N represent an estimate of the expected gain per customer that is correctly classified as insolvent and the expected cost per customer incorrectly classified as insolvent. Apparently, for small values of C_P / C_N predictions for the minority class must be very precise because it is too risky to misclassify majority cases and "precision rate" must be the leading performance measure. Conversely, for large values of C_P / C_N predictions must be very accurate for the minority class because it is very profitable to detect minority cases and the "true positive rate" is the leading performance measure. For in between values of C_P / C_N predictions must be a

combination of both because it is not so profitable to detect minority cases and less risky to misclassify majority cases and the "geometric mean of PR and TP" is the leading performance measure.

The decision for operationalizing the discovered knowledge is the last very important step in a KDD project with real enterprise data. This step involves integration of the results of the data mining process with the data available to the operators in the enterprise and the business knowledge that executive members carry, in a way that meets business objectives and supports strategic decisions. Using the design of a proposed architecture for an Intelligent Insolvencies Management System, we conclude that the key components of such a system are the IIMS Data Base, the Insolvency Predictor and the Heuristics Knowledge Base. The IIMS Data Base is built around the data model that results from the KDD process and incorporates data from the Corporate Information Systems with discovered knowledge. The Insolvency Predictor is a library of software components that activates the best classifier each time there is a need for predicting insolvency, based on current values of the cost parameters. Lastly, the Heuristics Knowledge Base is the heart of the Preventive Actions Support System, that supports the enterprise operators in taking preventive actions against suspected insolvent customers, evaluating previous actions and further filing business rules that have proved their effectiveness.

Overall, despite of the fact that this chapter has been inspired by a specific KDD project, the conclusions are generic enough and applicable to other projects that involve real enterprise data and the same data mining function.

References

R. J. Brachman and T. Anand, 1996, The process of knowledge discovery in databases: A human centered approach. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, Advances in Knowledge Discovery and Data Mining, chapter 2, pages 37-57. AAAI/MIT Press.

Brachman R.J., T. Khabaza, W. Kloesgen, G. Piatetsky-Shapiro, and E. Simoudis, 1996, Mining Business Databases, Communications of the ACM, 39(11), 42-48.

Cabena, P., Hadjinian, P., Stadler, R., Verhees, J., Zanasi, A., 1998, Discovering Data Mining: From Concepts to Implementation. Prentice Hall.

K. J. Cios and L. A. Kurgan, "Trends in data mining and knowledge discovery," Advanced Techniques in Knowledge Discovery and Data Mining, N. R. Pal, L. C. Jain, and N. Teoderesku, Ed. Berlin, Germany: Physica-Verlag (Springer), 2005, pp. 1-26.

Chan P., F. Wei, A. Prodromides, and S. Stolfo, 1999, Distributed Data Mining in Credit Card Fraud Detection, IEEE Intelligent Systems and Their Applications, 14(6), 67-74.

Chan, P. K. and S. J. Stolfo, 1998, "Toward Scalable Learning with Non-Uniform Class and Cost Distributions: A Case Study in Credit Card Fraud Detection", Proceedings Fourth Intl. Conf. On Knowledge Discovery and Data Mining, pp. 164-168.

Chawla, N. V., 2003, "C4.5 and Imbalanced Data sets: Investigating the effect of sampling method, probabilistic estimate and decision tree structure" Workshop on Learning from Imbalanced Datasets II, International Conference on Machine Learning, Washington D.C.

CRISP-DM, CRoss-Industry Standard Process for Data Mining, www.crisp-dm.org, 2001.

Daskalaki, S., I. Kopanas, M. Goudara, and N. Avouris, 2003, "Data mining for decision support on customer insolvency in telecommunications business", European Journal of Operational Research, 145(2), 239-255.

Daskalaki S., Kopanas I.. Avouris N., Machine Learning Techniques for prediction of rare events in a business environment, Proceedings 3rd Hellenic Conference on Artificial Intelligence, SETN 2004, pp. 79-88, Samos, May 2004.

Daskalaki, S., I. Kopanas, and N. Avouris, 2006, "Evaluation of classifiers for an uneven class distribution problem", Applied Artificial Intelligence, 20, 381-417.

Drummond, C. and R.C. Holte, 2000, "Explicitly Representing Ecpected Cost: An Alternative to ROC Representation", Proceedings of the sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 198-207.

Drummond, C. and R.C. Holte, 2004, "What ROC curves Can´t Do (and Cost Curves Can)", Proceedings of the ROC Analysis in Artificial Intelligence, First International Workshop, Valencia, Spain, August 2004, pg. 19-26.

Ezawa K.J., and S.W. Norton, 1996, Constructing Bayesian Networks to Predict Uncollectible Telecommunications Accounts, IEEE Expert/Intelligent Systems & Their Applications, 11(5), 45-51.

Fawcett T. and F. Provost, 1997, Adaptive Fraud Detection, Data Mining and Knowledge Discovery, 1, 291-316.

Fayyad U.M., G. Piatetsky-Shapiro, and P. Smyth, 1996, The KDD Process for Extracting Useful Knowledge from Volumes of Data, Communications of the ACM, 39(11).

Frawley W.J., G. Piatetsky-Shapiro, and C. Matheus, 1991, Knowledge Discovery in Databases: An Overview, in: Knowledge Discovery in Databases, G. Piatetsky-Shapiro and W.J. Frawley (Eds), AAAI Press/The MIT Press, Menlo Park, CA.

Jain, and N. Teoderesku, Ed. Berlin, Germany: Physica-Verlag (Springer), 2005, pp. 1-26.

Kalapanidas E., N. Avouris, M.Craciun and D.Neagu, Machine Learning Algorithms: A study on noise sensitivity, in Y. Manolopoulos, P. Spirakis (ed.), Proc. 1st Balcan Conference in Informatics 2003, pp. 356-365,Thessaloniki, November 2003.

Kopanas I., Avouris N.M. and S. Daskalaki, 2002, The role of knowledge modeling in a large scale Data Mining project, in I.P Vlahavas, C.D. Spyropoulos (eds), Methods and Applications of Artificial Intelligence, LNAI no. 2308, pp. 288-299, Springer-Verlag, Berlin.

Kubat, M. and S. Matwin, 1997, "Addressing the Curse of Imbalanced Training Sets: One-Sided Selection", Proceedings of the 14th International Conference on Machine Learning, 179-186.

Kubat, M., R. Holte and S. Matwin, 1998, "Machine Learning for the Detection of Oil Spills in Satellite Radar Images", Machine Learning, 30, 195-215.

Laurikkala, J., 2001, "Improving Identification of Difficult Small Classes by Balancing Class Distribution", in Artificial Intelligence in Medicine, S. Quaglini, P. Barahona, S. Andreassen (Eds.), LNAI 2101, p. 63-66.

Lewis, D.D. and W. Gale, 1994, A Sequential Algorithm for Training Text Classifiers, Proceedings of the Seventh Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, p. 3-12, Springer-Verlag.

Li Q., Li T., Zhu S., Kambhamettu C., (2002), "Improving Medical/Biological Data Classification Performance by Wavelet Preprocessing", In the Proceedings of ICDM 2002.

Provost, F. and T. Fawcett, 1997, "Analysis and Visualization of Classifier Performance: Comparison under Imprecise Class and Cost Distributions" Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, p. 43-48, Menlo Park, CA, AAAI Press.

Provost, F., T. Fawcett, and R. Kohavi, 1998, "The Case Against Accuracy Estimation for Comparing Induction Algorithms" Proceedings of the Fifteenth International Conference on Machine Learning (IMLC-98), pp. 43-48, Morgan Kaufmann, San Francisco, CA.

Provost, F. and T. Fawcett, 2001, "Robust Classification for Imprecise Environments", Machine Learning, 42, p. 203-231.

Riddle P. R. Segal, and O. Etzioni, 1994, "Representation Design and Brute-Force Induction in a Boeing Manufacturing Domain" Applied Artificial Intelligence, Vol. 8, p. 125-147.

Weiss, G. and F. Provost, 2001, "The effect of class distribution on classifier learning", Technical Report ML-TR-43, Department of Computer Science, Rutgers University.

Weiss, G. and F. Provost, 2003, "Learning When Training Data are Costly: The effect of Class Distribution on Tree Induction", Journal of Artificial Intelligence Research, 19, p. 315-354.

Wilson, T.D., 2002, The nonsense of 'knowledge management', Information Research, 8(1), paper no. 144, [Available at http://InformationR.net/ir/8-1/paper144.html]

Wirth, R., and Hipp, J., 2000, CRISP-DM: Towards a Standard Process Model for Data Mining, Proceedings of the Fourth International Conference on the Practical

Applications of Knowledge Discovery and Data Mining, pp. 29-39, Manchester, UK.

S. Zhang, C. Zhang, and Q. Yang, 2003, Data Preparation for Data Mining, Applied Artificial Intelligence, 17, No. 5-6, pp. 375-381.