SVM AND XBRL BASED DECISION SUPPORT SYSTEM FOR CREDIT RISK EVALUATION

Paulius Danenas, Gintautas Garsva

Vilnius University, Kaunas Faculty of Humanities, Department of Informatics, Muitines St. 8, Kaunas, Lithuania, paulius.danenas@vukhf.lt, gintautas.garsva@vukhf.lt

Abstract. This article presents a framework for an intelligent system for credit risk evaluation and a model for its implementation in cross-platform and database-independent way. This framework supports widely applied Support Vector Machines (SVM) technique as basis for intelligent evaluation and eXtensible Business Reporting Language (XBRL) standard which is widely developed, implemented and supported by governments and regulatory authorities. It also examines current SVM frameworks which can be used for new hybrid method development and the possibilities to implement them in a credit risk decision support system. A framework for a distributed credit risk evaluation system is presented together with its architecture and model for implementation.

Keywords: Support Vector Machines, SVM, XBRL, XML, artificial intelligence, machine learning, credit risk, evaluation, bankruptcy, framework, decision support system, DSS.

1 Introduction

Credit risk evaluation currently is a hot topic because of large number of companies which are going bankrupt, thus the price of mistake by approving credit request or rejecting it is high. This is why it is very important for the creditor to choose the correct principle or model for evaluation. Tools such as artificial intelligence and soft computing offer a possibility to create more sophisticated and precise models which can be implemented in such systems. Support Vector Machines (SVM) is one of these methods widely applied as an effective solution to many various pattern recognition, classification, regression and forecasting problems, including financial forecasting and credit risk evaluation. SVM technique has proven itself as an effective solution in credit risk field with results comparable to or better than most of other machine learning techniques [1]. SVM method has been combined with almost all popular natural computing techniques – fuzzy logic [2][3], Bayesian inference[4], genetic algorithms[5], rough sets [6], ant colony optimization [6], particle swarm optimization [7]. A research which combines SVM with widely used discriminant techniques to generate classification rules has also recently been made [8]. These investigations together with many others proved that SVM-based methods often outperformed similar techniques, such as Neural Networks, and can be deployed as a solution in credit risk field.

Expert systems and decision support systems (DSS) implementing these techniques are also widely used to help to solve this problem. They are widely researched and developed. There are many researches targeted to expert systems and their application in credit risk field; from rules-based using inference [9] to model-based with particular models [10][11], or are related more to specific tasks in credit risk field rather than for credit risk evaluation, e.g., credit card assessment [12]. Some researchers implement multi-agent based solutions as a network of problem solvers that perform together to solve problems [13], other focus on decision support system architecture with application of modern machine learning techniques such as SOM [14] or SVM [15]. Earlier developments also apply multiple criteria decision aid [16], as in most of similar researches financial statements and various financial ratios are used as main source of information. However, there seems to be a lack of research for integration of modern financial standards which offer new possibilities for financial and intelligent evaluation. The aim of this research is to propose a structure for decision support system for credit risk evaluation which implements advanced technologies and techniques such as Support Vector Machines (SVM) and eXtensive Business Reporting Language (XBRL).

2 SVM method and its implementations

SVM was created in the 7th decade by Vapnik in the labs of AT&T Bell Company. SVM and some other machine learning methods are a part of statistical learning theory, which describes the characteristics of self-learning machines. This theory gives theoretical foundations not only for SVM, but also for other important machine learning methods, such as neural networks. Vapnik defines SVM as learning machines that can do binary classification (pattern recognition) and real valued function approximation (regression evaluation). SVM implicitly maps m-dimension input data space to n-dimension possibility space where a linear classifier is created. The special ability is that at the same time it minimizes empiric classification error and maximizes geometric margin. This is the reason why this method is called maximum margin classifier.
The main task of binary classifier is to evaluate function $f: X \rightarrow \{\pm 1\}$ which maps input and output data. According to Vapnik (1995), SVM can be formulated as if empirical data $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \in \chi \times \{\pm 1\}$ is given with $x_i \in \mathbb{R}^n$, $y_i \in \{-1, 1\}$, the task is to find a decision function $f_{w, b}$ with the property $f_{w, b}(x_i) = y_i$, $i = 1..n$. The similarity measures for $\chi$ and $y$ can be formally described as a function $k$, commonly referred as kernel function: $k: \chi \times \chi \rightarrow \mathbb{R}$, which returns a numerical characterization of the similarity between $x$ and $x'$. It is used to transform input data space into another vector space (usually in much larger number of dimensions) in such a way that transformed data space can be separated in a linear way. The most popular and applied kernel functions are linear, polynomial, radial basis function (RBF) and sigmoid (as in other kernel methods, such as Neural Networks). SVM offers such advantages, according to Scholkopf [17]: similarity measures based on dot product in space F, interpretation based on analytical geometry, a possibility to select nonlinear mapping $\Phi$ for more suitable data representation.

There are many SVM algorithms and software for further classifier development available. Specialized knowledge in data mining and machine learning, as well as knowledge in particular field is required for creating hybrid models. A comparison of the capabilities of these frameworks is represented in Table 1. The comparison includes LibSVM[18], LS-SVM[19], also Lagrangian SVM[20], ASVM[21], SSVM[22], LPSVM[23] and Proximal SVM[24] by Mangasarian et al., SVMLight[25], B SVM[26], UniverSVM[27], SVM&KM Toolbox[28], SimpleMKL[29], mySVM[29], TinySVM[30], Core Vector Machines[32], PSVM[33], GPDT[34] and LIBLINEAR[35].

Table 1. Comparison of SVM implementations

<table>
<thead>
<tr>
<th>Problems solved</th>
<th>LibSVM</th>
<th>BSVM</th>
<th>UniverSVM</th>
<th>mySVM</th>
<th>SVM&amp;KM</th>
<th>SimpleMKL</th>
<th>SVMLight</th>
<th>CVM</th>
<th>GPDT</th>
<th>LIBLINEAR</th>
<th>Lagrangian SVM</th>
<th>ASVM</th>
<th>SSVM</th>
<th>LPSVM</th>
<th>Proximal SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Regression</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Ranking</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Clustering</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Feature selection</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

| Number of classes | One-class | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                   | Two-class | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                   | Multi-class | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |

| Programming language | JAVA | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                     | MATLAB | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                     | C/C++ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                     | Python | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                     | Perl | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |

| Kernel function | Linear | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                 | Polynomial | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                 | Sigmoid | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                 | RBF | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                 | ANOVA | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                 | Neural | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                 | Laplas | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                 | User-defined | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                 | Point | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                 | Sum | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
|                 | Inverted distance | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |

As SVM-based methods are widely applied in research, data mining and data analysis, SVM implementations are almost in every professional or scientific package for statistics or data mining (e.g., Weka[36] and RapidMiner[37], which have interfaces with popular SVM packages LibSVM, LIBLINEAR). However, currently there are no professional or scientific toolkits built exclusively for SVM method which would include all or most of the SVM frameworks mentioned in Table 1; the closest to the solution of this problem is Shogun[38] framework, but it is seems to be complicated to compile and use it on non-Unix OS. This
is important as different SVM classifiers might produce different results thus it might be useful to develop classifiers using several SVM methods.

3 XBRL standard

XBRL (eXtensible Business Reporting Language) is an XML-based standard for financial reporting currently becoming one of the most important frameworks in financial and banking sectors. This framework defines a widely used uniform and system-independent standard and utilizes the possibilities of XSL/XSLT/XSL-FO technologies, which can be applied to transform XML documents to RTF or PDF formats as well as other XML documents. It is applied as a standard in such countries as USA (Securities And Exchange Commission adopted it in 2005), UK (Financial Services Authority started using it in 2005). CEBS (The Committee of European Banking Supervisors) has also applied it as standard for 9000 credit institutions and investment firms to 25 supervisors/registrars[38]. This standard is also the basis of new SEC’s financial EDGAR database implementation. Great Britain will also apply XBRL based framework as a financial standard in 2011 April as all company tax returns will be bound to be filed online using the new iXBRL Standard[40].

XBRL is defined by two primary concepts: taxonomy and instance. Taxonomy defines all financial concepts that are used by a particular entity, as well as their inner relationships and internal or external resources; instance can be defined as the list of facts which has the structure defined in taxonomy. Semantic meaning is expressed by fully leveraging XLink to connect XML instance documents with any number of related XML fragments at a granular level using extended links. XBRL taxonomy consists of taxonomy and linkbase components; additional module specifications are also defined, starting from XBRL 2.1 version. These components are listed in Table 2.

Table 2. XBRL structure

<table>
<thead>
<tr>
<th>Component</th>
<th>Concepts stored (purpose)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Core</td>
<td></td>
</tr>
<tr>
<td>Taxonomy</td>
<td>Unstructured list of elements and references to linkbase files</td>
</tr>
<tr>
<td>Linkbase</td>
<td></td>
</tr>
<tr>
<td>Presentation</td>
<td>Relationships between elements in order to properly organize the taxonomy content (representation of the hierarchical relationships in particular business data)</td>
</tr>
<tr>
<td>Calculation</td>
<td>Definitions of basic validation rules for a particular taxonomy. Might be used to sort all monetary elements hierarchically (upper level entity is the result of operation with lower level elements)</td>
</tr>
<tr>
<td>Definition</td>
<td>Different kinds of relations between elements (distinguishing “general-special”, similarity “essence-alias”, requirement “requires-element”, tuple similarity “similar-tuples”)</td>
</tr>
<tr>
<td>Label</td>
<td>Labels for different languages</td>
</tr>
<tr>
<td>Reference</td>
<td>Pointers to source documents which describe the concepts defined in taxonomy</td>
</tr>
<tr>
<td>Additional modules (specifications)</td>
<td></td>
</tr>
<tr>
<td>XBRL Dimensions 1.0 [41]</td>
<td>Definition of additional structured contextual information for business facts in a manner similar to „dimension” concept in OLAP analysis. The base XBRL Dimensions specification defines three dimensions: reporting period, reporting entity (e.g.; a company or its division), and a loosely-defined reporting scenario, originally intended to distinguish between actual vs. projected facts. Taxonomies using XBRL Dimensions can define new dimensions as well as relate other taxonomy metadata (labels, presentation information, etc.) to them.</td>
</tr>
<tr>
<td>XBRL Formula 1.0 [42]</td>
<td>Definition of formulaic expressions using XPath 2.0 to validate XBRL instance information or to derive new XBRL facts (e.g. calculate secondary financial ratios)</td>
</tr>
<tr>
<td>XBRL Rendering [43]</td>
<td>Provision of documents which can be viewed in a browser while making use of XBRL tags and leveraging label and presentation linkbases to provide more comprehensive report definition, e.g., to specify exact relative location of particular concept</td>
</tr>
<tr>
<td>XBRL Versioning [44] (still in development)</td>
<td>Documentation of the differences between two versions of the same taxonomy (i.e., further taxonomy development, extension with new languages or business rules, errors found, laws and regulations that have changed, change of concepts or their removal and etc.). It should also provide ability to ensure compatibility for software vendors</td>
</tr>
</tbody>
</table>

SEC defines a particular taxonomy for Nationally Recognized Statistical Rating Organizations (NRSRO). It consists of 11 files and differs in absence of calculation linkbase, but additionally has type and role declaration schemas. The following aspects are defined in the latest version (currently Ratings 2009 Taxonomy): SEC category type (Corporate, Insurance, Finance, AssetBacked and Government), SEC subcategory type (Sovereign, US Public Finance, International Public Finance), Ratings item type (Program, Instrument, Shelf, Other), Instrument identifier scheme type (ISIN, SEDOL, VALOR, WKN, SICC, NRSRO, Other) and Issuer identifier scheme type (CISIP, DUNS, BIC, SICC, NRSRO, Other)[45].

Several public taxonomies are developed as authority standards; mostly applied are US GAAP used in USA as a standard for SEC reporting. Other well known taxonomies are COREP – Common Reporting taxonomy created by The Committee of European Banking Supervisors for solvency ratio reporting in Europe...
under future EU capital requirements regime and FINREP – Financial Reporting taxonomy created by The Committee of European Banking Supervisors, using Belgium Banking Taxonomy as basis. COREP is part of Basel II standard [38].

4 Proposed framework for credit evaluation decision support system

The human-system interaction is modeled by Use-Case model (Figure 1) which defines the main roles that use the particular functions. Two main roles are defined, Analyst and Risk Manager; however, it may depend on the structure or specifics of credit organization. The functions that they perform are targeted primarily at model creation and evaluation process as well as applying the results in practical activity and data management. The system is modeled as distributed system thus a modeling service subsystem is defined which represent the main classifier operations: model training (creation), testing and prediction. Model creation task is defined as a complex process which includes such subtasks as parameter selection (using stochastic or evolutionary techniques or selecting them manually), data preprocessing, training, testing and saving model to model repository. Data analysis can be viewed as a generic process that can be extended to statistical, financial or visual analysis. Yet analysis in credit risk is a complex task which needs all three ways of evaluation so this task can also be viewed as an aggregation of these three.

The structure of intelligent decision support system in presented in Figure 2. It has main components which were defined in earlier works (e.g., [10], [11], [14], [15]) – model repository, data store and user interface. However, additional possibilities are defined for data retrieval and processing as modern XML-based standards allow creating automatic data import from various sources.

Figure 1. A possible use case for credit risk evaluation system

The system’s structure is defined as consisting of three main layers – SVM based machine learning layer (further referred as SVM-ML layer) which purpose is to define and implement all the machine learning techniques and algorithms necessary for evaluation as well as other data mining tasks which need to be solved in machine learning process such as information processing, representation; data layer defining data that is available for modeling and stored in data storage facility; and credit risk evaluation layer (further referred as CRE layer) that implements whole analysis, modeling, forecasting and evaluation logic, as well as data visualization. The separation of these aspects gives a possibility to use machine learning techniques implemented in this system to solve other problems by implementing only the logic specific to these problems. The main aspects of this system are also defined as particular layers:

Data source interaction layer – it is defined in both SVM-ML and CRE layers. SVM-ML layer interaction sublayer includes database interaction layer with object persistence frameworks (such as Hibernate) and database connection frameworks (drivers), as well as data standards commonly used in machine learning software (such as Weka ARFF, Comma Separated Values (CSV) formats or interoperable Predictive Model Markup Language (PMML) standard). It also defines the interfaces for intelligent information retrieval (not necessarily financial) using Web Services or intelligent agents. CRE layer additionally defines (or can be thought as an extension of previously described layer) financial standards and data sources specifically for finance or credit risk related tasks. It also has a mapping package that contains the mappings between XBRL (or other standards) and data stored in Data Layer. The XBRL instances might be entered manually or automatically,
by using, e.g., RSS feeds which contain links to these instances thus automating data retrieval process. RSS is also proposed by SEC as an option[45].

Source: created by authors, using [11][14]

Figure 2. The structure of intelligent credit risk evaluation DSS

Information Processing layer – also defined in both SVM-ML and CRE layers. It implements main tasks that are solved during the whole intelligent model creation process before training using one of the algorithms. It defines such standard data mining aspects as information retrieval, data extraction and cleansing, data transformation (e.g., using Principal Component Analysis, Independent Component Analysis, Factor Analysis, etc.), normalization/standardization, data imputation. The same layer defined in CRE layer implements specific tasks, e.g., specific transformations, data transformation to absolute or percentage changes between particular ratios during particular period and etc.

Data Layer – defines all the data that is stored in data store (multidimensional data warehouse, database or other source). The system described here uses company data, financial data (data extracted from financial reports), company management and personnel data, historical records, market data, also macroeconomic and statistical data for macroeconomic environment evaluation (this type of data for analysis is also defined in earlier works, e.g., [10], [11] or [14]). It also contains metadata, such as reference or multilanguage data, as well as financial ratings and historical information. As it was mentioned before, ratings data can also be retrieved from XBRL documents with ratings data (e.g., instances of SEC RATINGS taxonomy). The last component is a model repository which contains all intelligent, statistical or other models (including SVM-based or hybrid models), as well as their execution log, evaluation results and their metadata.
**Representation layer** - this module includes all methods and operations which are used for representation and visualization of results. It is more generic in case of SVM-ML (defines standard representations of training, testing and prediction results as well as their visualizations). CRE layer defines more sophisticated modules such as OLAP analysis together with representation of financial analysis, simulation/modeling and forecasting as well as data management functionality;

**Financial Analysis, Modeling and Forecasting** modules are defined particularly for risk evaluation layer as they present analytics, simulations and forecasting of particular domain. Here the list of methods is not complete; it can be extended by using various modern fields and methods of computational finance.

### 4.1 A possible implementation scenario

Figure 3 represents an UML implementation diagram for DSS based on proposed framework. It describes all server nodes, execution environments as well as possible technologies that can be used to implement the described system. As Table 1 shows, SVM frameworks are implemented in different languages. This makes a difficult task to combine them in hybrid algorithms or use together in single system. Thus a framework for interoperability, such as CORBA, COM or Web Services, has to be implemented in this system to ensure that as many SVM implementations might be used as possible; as it was mentioned in related work, different SVM algorithms often show different results. Web Services was selected to implement the SVM classifiers as services in this case, as it ensures maximum compatibility and is easier to implement than CORBA because known CORBA open source implementations seem difficult to apply (i.e., they do not have good reference material covering all aspects or are difficult to implement in cross-platform manner), and COM is not cross-platform. Three operations most commonly used in classification tasks are defined in WSDL document: Training, Testing and Prediction. Yet, it can be extended with other operations, as some SVM implementations also offer additional functionality, e.g., ranking (SVM-Light) or outlier detection using one-class SVM.

JAVA was chosen as an implementation language for the whole system as it offers many possibilities and frameworks needed to implement functionality described here; e.g., WEKA and RapidMiner can be a good choice to implement Machine Learning layer functionality as they contain implementations of mostly referenced SVM algorithms together with many others. There are several known open source cross-platform JAVA XBRL implementation frameworks (xBreeze Open Source Edition provided by UBMatrix, Batavia XBRL Java Library, XBRLAPI.org) which would allow to develop XBRL Import Component. It is also useful to store retrieved XBRL data in XML native database which also provides search and querying facilities and can be used as storage facility of original XBRL instances. A similar component might also be implemented for other similar
XML-based formats, e.g., MDDL* and SDMX** standards might be utilized for macroeconomic data import, RIXML*** - for research related data import (depending on the subset defined in this structure). JAVA also offers good possibilities for enterprise-level development and integration with other systems (e.g., using ESB and JMS for messaging-oriented integration development) as well as web interface implementation and development of Web services. If SVM classifier cannot be implemented in JAVA then C++ can be a good option as Table 1 shows that most of them are written in C/C++ or MATLAB (using its mex compiler). These aspects are defined in proposed system framework. Such features of implementation proposed in Figure 3 can be emphasized:

- Cross-platform – XBRL and other XML-based standards do not depend on any system platform; JAVA can be deployed on any Windows/*nix/BSD platform and C++ also has popular cross-platform frameworks for development (e.g., Qt framework by Nokia);
- Database-independent – application of Hibernate object persistence technology allows implementing the system almost independent on DBMS as SQL queries in code are expressed using Hibernate’s own HQL technology and then translated to corresponding SQL dialect.

4.2 Comparison with similar systems and possible future extensions

Tsaih et al. proposed an N-tier architecture with internal credit scoring model transformation into XML document. It consists of thin client layer (representing GUI in Web browser), middle tiers include the web server, Management Application Server (MAS), which provides interfaces to manage TDV, Database and XML repository and perform other management tasks, Loan Processing Subsystem (LPS) with Case Processing Application Server (CPAS) and Evaluation Module (EM) with XML parser and model engine sub-modules, and Model Installing Subsystem (MIS) which consists Model Defining Application Server (MDAS) and Model Recording Module (MRM) [47]. The structure of this prototype is quite similar to proposed system’s structure, although it is targeted at real world transaction processing whereas our model is more suitable for credit risk modelling and analysis. However, its extension with industrial modules for loan processing and banking operations might be considered in the future. PMML standard might be considered as better approach for model storage, transfer and manipulation, as it offers a standardized approach and is maintained and supported by many vendors and organizations [48]. Kotsiantis et. al. developed a distributed ontology-based credit evaluation system with an application of C4.5 algorithm for scoring and intelligent search and reasoning possibilities[46]; although they referred to XBRL as one of the options which would enable analytical possibilities offered by Semantic Web technologies they chose their own developed ontology to represent financial statements. Both of proposed system prototype solutions were engineered using JAVA technologies which prove to be a good choice for implementation of such system. This framework can be extended as XBRL standard together with modern Semantic Web technologies offers many possibilities, e.g., automation of information retrieval and model updating in real time thus making evaluation even more precise. Another important extension would be business rules integration (they are defined as an artifact in Data Layer). The rules defined in XBRL Calculation Linkbase might be leveraged to ensure the integrity and validity of data as well as define additional secondary financial ratios.

5 Conclusions

Credit risk evaluation is a sophisticated task which includes many aspects, views and approaches which should be used by analyst to properly assess customer’s abilities and probability of default. Many methods are applied in this field and intelligent hybrid models based on SVM have proven themselves to be more precise and effective. A proper structure of system implementation needs be selected which would allow to implement, test and use intelligent models in credit organizations. This article proposes a structure for such system with such properties as cross-platform and database-independent. This framework is also integrated with XBRL standard which is implemented by such authorities as Securities Exchange Commission (SEC) and European Union, and is part of Basel II requirements. This standard also offers new possibilities for further research, thus, this framework might be extended with new abilities in the future.

References


* Market Data Definition Language (MDDL) (http://www.mddl.org)
** Statistical Data and Metadata Exchange (SDMX) (http://www.sdmx.org)
*** Research Information Markup Language (RIXML) (http://www.rixml.org)


[27] UniverSVM[interactive, 2010.11.30], access via the Internet http://www.kyb.mpg.de/fs/people/fabee/universvm.html


