Informatics in Education, 2015, Vol. 14, No. 1, 13–34

DOI: http://dx.doi.org/10.15388/infedu.2015.02

Application of Interactive Classification System in University Study Course Comparison

Ilze BIRZNIECE[1], Peteris RUDZAJS[2], Diana KALIBATIENE[3], Olegas VASILECAS[3], Edgars RENCIS[4]

[1] *Riga Technical University, Study Department*
Kalku street 1, Riga, LV-1658
[2] *Riga Technical University, Faculty of Computer Science and Information Technology*
Institute of Applied Computer Systems
Kalku street 1, Riga, LV-1658
[3] *Vilnius Gediminas Technical University, Faculty of Fundamental Science*
Department of Information Systems
Saulėtekio al. 11, LT-10223
[4] *Riga Technical University, Institute of Telecommunications*
Azenes street 12, Riga, LV-1048
e-mail: {ilze.birzniece, peteris.rudzajs}@rtu.lv; {diana.kalibatiene, olegas.vasilecas}@vgtu.lt; edgars.rencis@lumii.lv

Received: June 2014

Abstract. The growing amount of information in the world has increased the need for computerized classification of different objects. This situation is present in higher education as well where the possibility of effortless detection of similarity between different study courses would give the opportunity to organize student exchange programmes effectively and facilitate curriculum management and development. This area which currently relies on manual time-consuming expert activities could benefit from application of smartly adapted machine learning technologies. Data in this problem domain is complex leading to inability for automatic classification approaches to always reach the desired result in terms of classification accuracy. Therefore, our approach suggests an automated/semi-automated classification solution, which incorporates both machine learning facilities and interactive involvement of a domain expert for improving classification results. The system's prototype has been implemented and experiments are carried out. This interactive classification system allows to classify educational data, which often comes in unstructured or semi-structured, incomplete and/or insufficient form, thus reducing the number of misclassified instances significantly in comparison with the automatic machine learning approach.

Keywords: machine learning, interactive classification, inductive learning, curricula comparison.

1. Introduction

The growing amount of available information in the world encourages the use of automatic data processing techniques that reduce human routine work. This is the place for artificial intelligence and its subfield machine learning. Education belongs to areas where extensive data exploration is needed. The research is focused on the study course compatibility analysis in higher education. The comparison of study programmes and courses is necessary in several educational tasks. One of them is student mobility. Taking into consideration the number of different education institutions operating inside the global knowledge provision space this is a time consuming task. Although one of the main features of the Bologna process is to encourage creation of a common model for Higher Education in Europe (Kennedy et al., 2009), there still does not exist a generally established standard for describing study courses in all universities, and they currently appear both as semi-structured and unstructured textual descriptions. This fact creates the main difficulty for course comparison automatically. Therefore, in reality comparison of study programmes and individual courses is a task that is performed manually.

Application domains are getting more complex in terms of data amount, representation forms, relationships within data etc. For this reason machine learning approaches face new challenges in solving tasks which could benefit from automated solutions but do not conform to typical machine learning application areas. Classification is one of the machine learning tasks where the program learns to predict class label of new instances from a human or environment provided facts. Classification process can be devidied into classifier building (or training), testing and applying steps. From all range of classification approaches we consider inductive learning algorithms in a form of decision trees and rules. They are widely used in machine learning tasks and hold a strong position as reliable classification methods that can explain the way how the decision is being made (Aksoy, 2008). In computer science, inductive learning is learning by example, where a system tries to induce a concept description $c: X \rightarrow L$ from a set of observed instances $X = \{ x_1, \ldots, x_i \}$ with a known set of class labels $L = \{ l_1, \ldots, l_j \}$. Each instance x consists of attribute-value pairs $\{(a_1, v_{a1}), \ldots, (a_n, v_{an})\}$.

This work can be characterized as applied, experimental and quantitative research. It is aimed at developing an automated or semiautomatic classification solution which incorporates both machine learning facilities and interactive involvement of a domain expert in the classifier's applying stage for improving its results if the classifier makes uncertain classification. The rest of the paper is organized as follows. Research objectives and related work from both educational and machine learning aspects are given followed by interpretation of study course comparison task in machine learning context. We describe contents of developed Interactive Classification System's (InClaS) framework. An instance of such system is built and used to carry out study course comparison empirically. Experimental settings, achieved results and conclusions sum up this research paper.

2. Research Objectives and Related Work

Motivation of the reseach and development on interactive inductive learning based classification system comes from several sides. One of them is inappropriateness of the automated classification methods for all domains where machine techniques could be applied to. Other facilitator for developing an interactive classification system is the practical need in the area of curricula comparison. We will discuss both of these issues briefly.

Nowadays information is often organized in complicated forms for machine learning, like plain (unstructured) text, graphs, semi-structured text, etc. The transformation from the original data to the classifier-acceptable data structures is needed, and in this process some information can get lost or mapped inaccurately. This leads to creation of an incomplete classifier that does not generalize well the problem domain and probably will not be able to make predictions for all new unseen instances when the classifier is applied. We state that the solution for this problem is creation of a semi-automatic classification system to give the expert a wider control over the classification process and use his/her knowledge for gradual improving of it.

Considering educational document comparison, a term *educational document* is used to denote different types of materials for educational content and assessment, including course descriptions, teaching materials, academic credentials, etc. The necessity to compare educational documents appears in different forms and can be conditionally divided into three categories (Alves and Figueira, 2011; Anohina-Naumeca *et al.*, 2012; Biletskiy *et al.*, 2009; Biletska *et al.*, 2010; Ranganthan *et al.*, 2006; Rudzājs and Kirikova, 2012, 2009; Dagienė *et al.*, 2013; Teodosiev and Nachev, 2012). These categories are:

(1) Student exchange programmes.
(2) New curriculum development.
(3) Teaching material and learning object categorization for, e.g. e-learning systems.

In the scope of this paper we consider only the first category and target mutual comparison of course content.

Examples of study course textual descriptions are given in Fig. 1 to demonstrate their variety.

Although there are attempts to put it this way, study course comparison does not fully belong to the problem of text classification. Study course description most often is a semi-structured text which usually includes sections like “prior knowledge”, “learning outcomes”, etc. It is important to distinguish between these sections. Besides, a semi-structured text has a significantly richer and more complicated structure than a plain-text, and the relation among semi-structured documents is harder to be fully utilized if only text categorization is used (Sebastiani, 2002; Jianwu Yang and Chen, 2002; Müller, 2010). There could help the study course comparison approach which uses formalized semantically meaningful attributes and interactive approach.

Existing research in the area of curricula comparison does not solve the problem of study course comparison. It has been proposed to represent study programmes as concept maps, and a system based on schema matching (Saleem *et al.*, 2008) of concept

RTU Course "Knowledge Management Systems"

12307 Sistēmu teorijas un projektēšanas katedra

General data

Code	DSP701
Course title	Knowledge Management Systems
Course status in the programme	Compulsory/Courses of Limited Choice
Course level	Post-graduate Studies
Course type	Academic
Field of study	Computer Science
Responsible instructor	Kirikova Mārīte
Academic staff	Apšvalka Dace
Volume of the course: parts and credits points	1 part, 4.0 Credit Points, 6.0 ECTS credits
Language of instruction	LV, EN
Possibility of distance learning	Not planned
Abstract	In this course students will learn about the concepts of organisational learning and knowledge, essential factors of organisational learning, knowledge flow and networks and technologies supporting them. Human-computer interaction and interface design will be discussed. Students will learn to define knowledge management strategy, to design knowledge management systems, to plan the development of these systems and will be familiar with different knowledge management technologies.
Goals and objectives of the course in terms of competences and skills	Successful completion of this course will provide students with the content and skills necessary to: explain the impact of the nature of knowledge on the management of knowledge; understand and interpret the concept and objectives of knowledge management in terms of advanced business practices and technologies; analyse knowledge processes within an organisation in terms of organisational performance and development; identify approaches (tools and techniques) that organisations may take to make a contribution to organisation's knowledge processes; understand the need for equal consideration of technological, human and organisational aspects; identify and define the best approach of knowledge.
Structure and tasks of independent studies	In individual assignments students will explore and analyse knowledge management solutions

MBI 665 **Knowledge Management and Decision Support**

This course introduces students to knowledge management practices and the technologies collectively called decision support systems. To cover the most current topics affecting how individuals and organizations use computerized support in making decisions. Business applications of data warehouses, online analytical processing, group support systems, knowledge acquisition and representation, knowledge management, knowledge-based decision support and intelligent systems will be explored.
PREREQ: MBI 625 MBI 625

Fig. 1. Study course description in Riga Technical University (`http://www.rtu.lv/`, 2013) (top) and Northern Kentucky University (`http://www.nku.edu/`, 2012) (bottom).

maps has been developed (Anohina-Naumeca *et al.*, 2012). In this approach curricula are compared according to their structure. However, one of the basic tasks in comparing curricula is the comparison of individual courses in the course content level that has not been included in this research. Unsupervised classification mechanism presented by (Alves and Figueira, 2011) organizes educational documents from e-learning system into clusters. Design of (Ranganthan *et al.*, 2006) describes methodology for classification of learning objects which can appear in different forms, e.g. course outlines and transcripts without well-defined metadata. Classification of a new learning object is done by finding the smallest distance to the cluster, where clusters define subdomains of interest. However, in (Alves and Figueira, 2011) and (Ranganthan *et al.*, 2006) it is not the course description that is used as the input. Both of these approaches also assume that objects relate to only one category, although in practice it is not the case when comparing documents in distinct curricula.

As claimed in (Coletta *et al.*, 2011), comparative analysis of educational documents is a complicated task both for experts and computer systems. Therefore automation of this process requires specific approaches and expert participation. Semi-structured document representation requires the use of various information extraction methods. Authors of Academic e-Advising system (Biletskiy *et al.*, 2009) point out that system's results could undoubtedly be improved by expanding the size of the training corpora and involving an expert. The system would also benefit from the implementation of an easy mechanism for manual inspection and augmentation of the extracted data to improve data quality for further use.

Analysis of related works shows that educational document comparison requires automated but not automatic approach to receive reliable results. It is also worth noting that despite the fact that different study programmes do not have the same granularity and content distribution between courses (Biletska *et al.*, 2010; Ranganthan *et al.*, 2006), the course similarity has been considered only using one-to-one correspondence. None of the presented systems so far deals with the possible one-to-many correspondences between courses or uses multi-label classification approach. Although the need for expert involvement has been emphasized, methods used so far do not foster collaboration with an expert.

A multi-label class membership requires the use of appropriate and more sophisticated classification methods. Multi-label classification is useful in practice, when an object naturally belongs to more than one category (Thabtah *et al.*, 2005). In multi-label classification, examples are associated with a set of labels $Y \subseteq L$ where L is a set of labels in contrast with the traditional single-label classification where examples are associated with a single label l from L, $|L| > 1$.

As the classification task in course comparison context is complicated because of insufficient amount of training examples and possibly incomplete formalized study course descriptions, the automatic classifier may not make enough informed decision on its own. It may happen that none of the classification rules fit the new instance when the classifier is applied. There are several methods to deal with this problem. Inductive learning systems with a low number of unclassified instances usually apply a default rule for classifying new instances that none of the rules in the rule base can classify (Clark and Niblett, 1989). A default rule comes from CN2 (Clark and Niblett, 1989) and AQ (Michalski *et al.*, 1986) algorithms and predicts the most common class in a particular data set. If a data set contains many classes and, moreover, if all of them occur equally frequently, assigning one certain class to all unclassified instances will not lead to a high accuracy of the classifier. Even more, most of nowadays classification algorithms do not admit their inability to classify instance but classify it anyway (correctly or incorrectly) making it harder for the system's user to detect the boundary of "real knowledge" of the classifier. Therefore, the interactive semi-automatic approach which takes into account confidence with which the classifier makes its decision should be used.

The authors of this paper have analysed and summarized a number of papers (Okabe and Yamada, 2002; Tanumara *et al.*, 2007; Buntine and Stirling, 1991; Hadjimichael and Wasilevska, 1993; Wong and Laung, 2000; Li *et al.*, 2009) referring to the concept *interactive inductive learning* and exploring the idea of user interaction in a concept learning process. Depending on phase in the classification process where a human interaction is expected, a diagram for abstract comprehension of different existing approaches to the interactive inductive learning has been created (see Fig. 2). In the stage of classifier building, data is passed to the learning algorithm (phase A) and classification rules are given to output (phase B). In the stage of classifier ap-

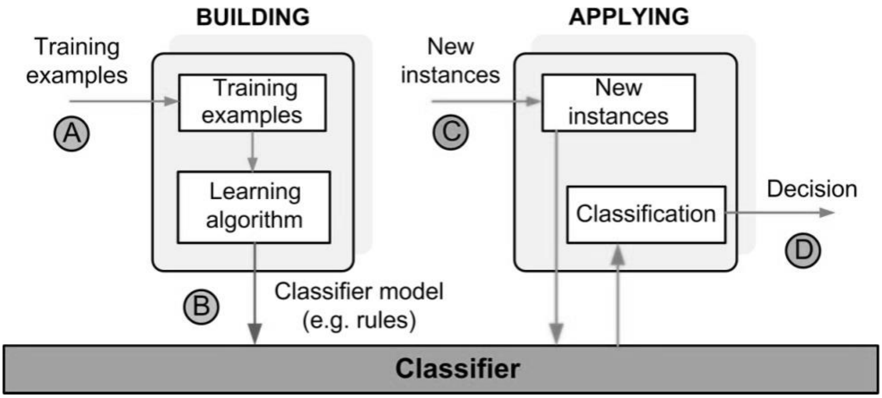


Fig. 2. Phases when an expert can interact with the classifier.

plying, a new instance (instances) with no classification is provided to the classifier (phase C) and a decision of its class is expected to be received (phase D). Methods described in aforementioned works provide interaction with an expert in phases A, B or D which is either too early or too late to handle new instances that the classifier cannot classify, but not in phase C when a particularly hard-to-classify instance arrives. Special methods of interactive classification – active learning (Settles, 2010) and Ripple Down Rules (Brian and Compton, 1995) – have been also considered. In our research we are dealing with phase C.

According to the presented related work, target of the research is defined as development of *inductive learning based interactive multi-label classification system for supporting study course comparison*.

3. Interpretation of the Course Comparison in Machine Learning Context

According to the related work the problem domain – university study course comparative analysis – can be defined by the following *features which intended machine learning solution should take into account*:

- Understanding decision making steps is important for the classifier's user and the expert.
- Available initial learning base (in this case – expert-made course comparisons) is small.
- Initial data (textual course descriptions) is semi-structured or unstructured.
- Domain defines many classes (course labels) with equal frequency.
- Each object (study course) can have a multi-label class membership (correspondence).

This subsection clarifies the study course comparison as a classification task. To do it, we need to define attributes and classes. Study course description does not naturally possess well-defined attributes. To apply inductive learning or other classification methods, a formalized attribute-value based representation is to be achieved. For practical implementation of university course comparison two main settings are chosen – direct and indirect comparison. For direct comparison, text classification approach is applied which makes use of word vectors obtained from full course descriptions. Indirect comparison involves mediating framework for extracting semantically meaningful information from course descriptions. Meaningful and usually accessible course attributes are learning outcomes, study level and the number of credit points. Learning outcomes can be described in different ways; hence, a need for unification arises. European e-Competence Framework (e-CF) (*European E-Competence Framework*, 2012) is chosen as a mediating framework because it is European-wide framework and is oriented to learning outcomes that are important for course comparison.

For the training set, an expert defines classes (i.e. detects correspondences) to unknown study courses. Note that the expert can assign more than one class since the courses can overlap in their content.

Fig. 3 demonstrates an overview of way for achieving formalized course attributes and detected classes in direct and indirect comparison. Formalization is done in order to prepare appropriate input data format for classification algorithms.

It is worth noting that the attribute selection in this task is not predefined. Data sets extracted in direct and indirect comparison are used separately; therefore, practical experiments can demonstrate the classifier's ability to generalize from provided attributes in both representations and provide a justification for preferring one or another.

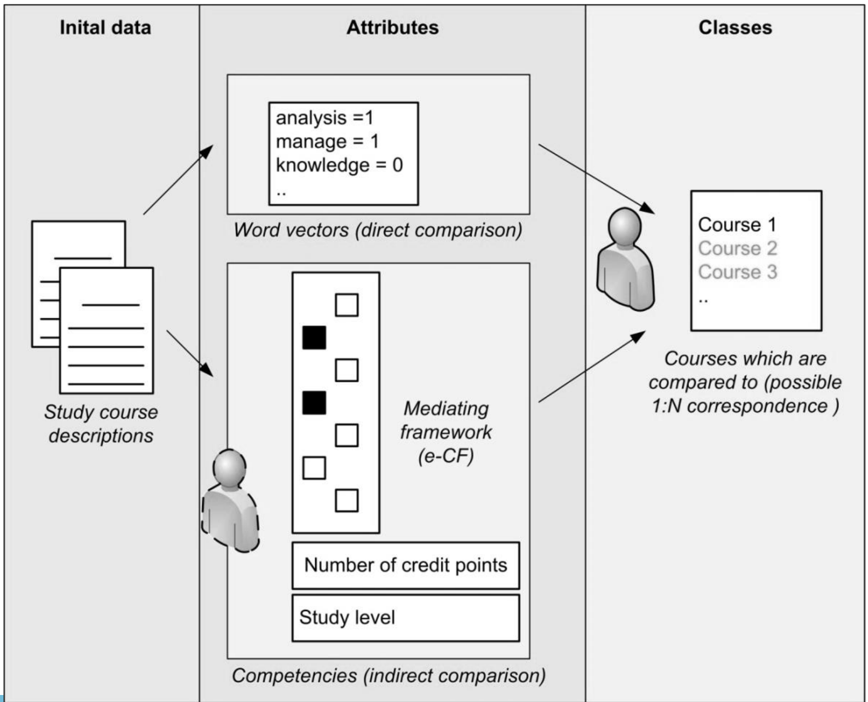


Fig. 3. Approach for formalizing study course comparison.

4. Framework of Interactive Inductive Learning Based Classification System (INCLAS)

The proposed framework is developed to define how to create interactive classification system for particular implementation. Various components extending traditional classification system are designed (Birzniece, 2010; Birzniece and Kirikova, 2011; Birzniece and Rudzajs, 2011). These components are extended and amalgamated in the **In**teractive Inductive Learning Based **Cla**ssification **S**ystem's (**InClaS)** framework. Fig. 4 depicts three levels of this framework which are explained in short afterwards in this section.

4.1. *Generic Model*

InClaS generic model consists of the components as follows (see also InClaS generic model in Fig. 4):

- General scheme of interactivity.
- Definition regarding an *uncertain classification*.
- Interactive classification system's structure, its modules and connections.
- Suggested approaches for updating the classifier.

There are also parameters identified which are to be determined in each InClaS application area (see Fig. 4). The choice of a learning algorithm and its settings as well as descriptive attributes is to be made in all classification tasks, and the interactive approach makes no difference in this aspect.

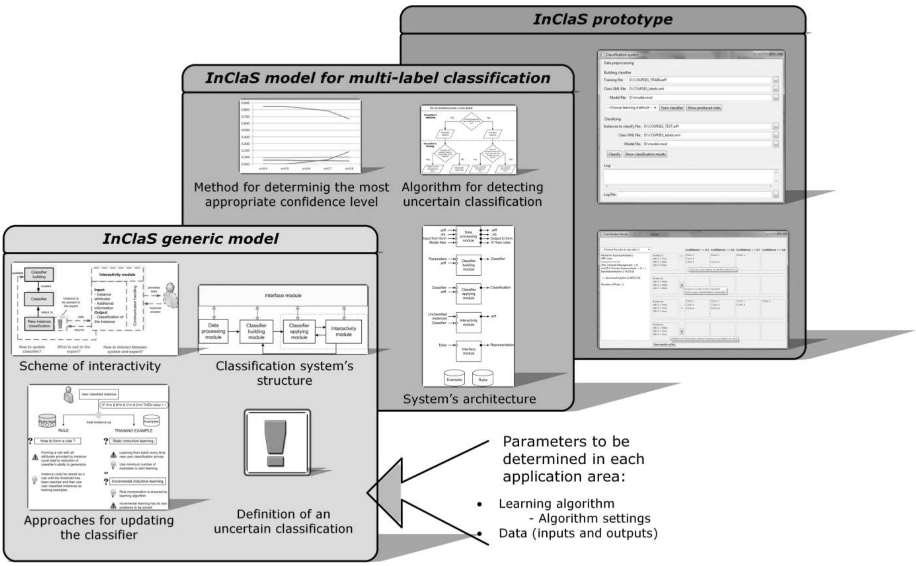


Fig. 4. Framework of InClaS.

4.1.1. *General Scheme of Interactivity to be Implemented*

Fig. 5 shows how the interactivity is implemented into the general model of the classification process. Blocks with solid line are elements of a traditional automatic classification system. Blocks and arrows with interrupted lines are introduced to ensure interactivity with a human expert in order to assign class value(s) for uncertainly classified instances. This includes the following functions:

1. Capturing uncertain classifications in the classifier applying stage.
2. Forwarding these instances and additional information to the expert.
3. Receiving and processing the expert's decision.
4. Using expert-provided knowledge to update the classifier.

The questions which arise from the classification system's extension with interactivity are resolved within the next subsections that concern other components of InClaS generic model.

4.1.2. *Definition of an Uncertain Classification*

To answer the question "What to ask to the expert?", it is important to define the characteristics of instances which are uncertain to the classifier and could benefit from the expert's perusal. Therefore, notion of terms used variously in machine learning literature – *unclassified instance*, *instance with low classification confidence* and *uncertain classification* – are clarified and their meanings in the context of this research are defined for further use.

Unclassified instance is an instance which was not covered by any rule (or a corresponding leaf in the decision tree) from the classifier's model in the classifier applying stage.

Taking into consideration the *confidence which the classifier associates with the* rule (or leaf) that is used to classify an instance, the classifier's decision can be marked as not confident enough. Confidence is based on example distribution in the training set which was used to build the classifier. An instance is said to be classified with low confidence if the confidence level for the class assigned by the classifier is below the selected threshold.

Uncertain classification includes both of above-mentioned aspects and is a term used to ascertain either unclassified or with low confidence classified instances.

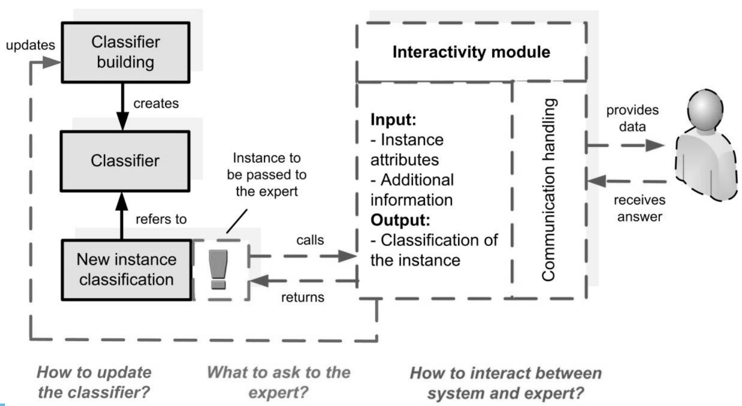


Fig. 5. Framework of InClaS.

Regarding multi-label classification more sophisticated uncertain classification definition is to be applied since more than one class can be assigned to an instance. This aspect as well as the method of achieving the most appropriate confidence levels for different data sets is outlined in next section along with InClaS particularization for multi-label classification tasks.

4.1.3. *Interactive Classification System's Structure*

The system's structure holds part of the answer to the question "How to interact between the system and the expert?". A modular structure is chosen for the interactive classification system. Fig. 6 shows actions typically performed in the interactive classification system, without the inner process details within modules. The user can provide data for classifier training (1a), initiate classifier building (2a) and submit new instances to be classified (3a). If the classification can be made by rules in the Classifier, the user receives classification results as a response (3c). If there is an instance which cannot be certainly classified by the Classifier applying module, a request to the Interactivity module to handle the situation is sent (3d). The Interactivity module asks for an expert classification of the instance through interface (3e); this is the situation when a request for a response is being sent from the system to the user, not vice versa. After receiving the expert's feedback, the Interactivity module informs the user and updates the Example base with a new example that was built from the instance and the user-given classification to it (3g). Consequently, the Classifier can be updated. Techniques which an expert can use for decision making regarding instance classification are not considered in the scope of this work. The classification system accepts a single expert opinion.

4.1.4. *Suggested Approaches for Updating the Classifier*

To answer the question "How to update the classifier?" activities for accepting the expert's classification and updating the classifier in response to this decision are defined.

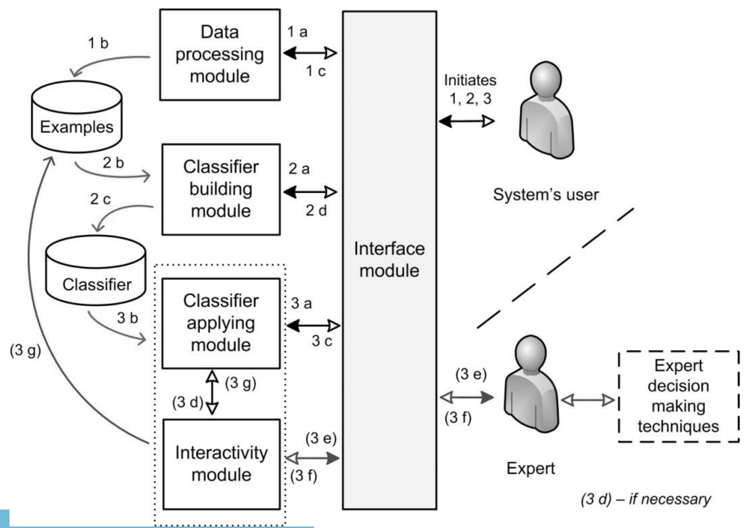


Fig. 6. Modules and main processes within the interactive classification system.

The task of the interactive system is to accept the expert's decision and to update the classifier in response to this decision. The main considerations are either to treat the expert's classified instance as rule or use it as a training example. As a result, two approaches for expert-made decision incorporation into the classifier, which maintain consistency of the classifier, are identified – *Incremental learning approach*, which uses one of readily available incremental learning algorithms and authors proposed *Threshold based static learning approach*. Both approaches are described in detail in (Birzniece, 2010).

InClaS generic model provides general-purpose components to develop either a single-label or a multi-label classification system. Due to the scope of problem to be solved, InClaS is further developed to serve classification tasks with multi-label class membership. It is described in the next section.

4.2. *InClaS Model for Multi-Label Classification*

To deal with multi-label classification, InClaS model has been extended with the following additional and specified components. See also InClaS model for multi-label classification in Fig. 4 and detailed descriptions of components in (Birzniece, 2010; Birzniece and Kirikova, 2011; Birzniece and Rudzajs, 2011). They are as follows:

- Algorithm for detecting uncertain classification.
- Method for determining the most appropriate confidence level.
- Architecture of a classification system.

4.2.1. *Algorithm for Detecting Uncertain Classification*

Multi-label class membership requires an extended definition of uncertain classification and unclassified instance since each object can belong to an unknown number of classes which makes the classification task more complicated. One of widely used approaches for multi-label classification is binary relevance (Tsoumakas and Katakis, 2007), where it is suggested to split the initial problem into several single-label classification tasks. Therefore, the classification of a new instance comes from a combination of *n* single-label classifiers where each classifier predicts classification for just one of all *n* classes. If none of the classifiers predicts positive class, instance is defined as unclassified (thus also assigning uncertain classification mark). An algorithm for detecting uncertain classification in multi-label domains defines that an instance is uncertainly classified if at the chosen (or default) confidence level none of actual classes of instance is predicted.

To consider usefulness of user involvement in classification process and impact to number of misclassified instances the authors of this paper introduce several simple measures to be detected and evaluated later in experimental phase:

- *Partly correct or completely correctly classified instance* (*PC*) – at least one of predicted classes is the actual class of an instance, $Y_i \cap Z_i \neq \emptyset$, where Y_i – actual label set of instance i, Z_i – predicted label set of instance i.
- *Misclassified instance* (*M*) – none of predicted classes is the actual class of an instance, $Y_i \cap Z_i \neq \emptyset$.

- *True uncertain classification* (*TU*) – the classifier would misclassify an instance (*M*) (that is, with the confidence level 0.5 none of actual classes would be predicted).
- *False uncertain classification* (*FU*) – the classifier would classify instance partly or completely correctly (*PC*) (that is, with the confidence level 0.5 at least one of actual classes would be predicted).

Certainly, it is desirable to strive for a classifier which maximizes the number of *PC* instances; however, if achieving high number if *PC* instances is hindered due to incompleteness of the classifier, e.g. because of small training set, the classification system should at least be aware of its "lack of knowledge" and be able to detect uncertain classifications.

4.2.2. *Method for Determining the Most Appropriate Confidence Level*

It is assumed that a higher confidence level brings less misclassified instances, although it increases the number of uncertain classifications (instances below this confidence level) which in the interactive approach are passed to the expert. Therefore, the compromise should be achieved between the expert's workload and the number of misclassified instances left in the classification results. Different domains have various specifics regarding the confidence level. Both manual and automatic method for determining the most appropriate confidence level for each data set have been developed to address this issue (Birzniece, 2013). The goal of the method is to determine the most appropriate confidence level where number of misclassified instances (*M*) is minimal taking into consideration given constraints regarding the expert's workload.

4.2.3. *Architecture of an Interactive Multi-Label Classification System*

Design of an interactive inductive learning based classification system for a multi-label classification task is guided by a five step procedure for designing intelligent systems by (Bielawski and Lewand, 1991). Design decisions for a university study course comparison task are explained resulting in a more detailed system's structure which defines particular inputs and outputs of the modules. This component of the InClaS model is detailed in the authors' publications (Birzniece, 2011; Birzniece and Kirikova, 2011; Birzniece and Rudzajs, 2011).

The developed InClaS generic model and its extension for multi-label classification provide a sufficient theoretical and methodical ground for implementing an interactive classification system as a software prototype.

4.3. *InCasS Prototype*

This subsection describes the main functionality of the prototype, paying attention to embodiment of InClaS model components into software. Data input and output is provided through graphical user interface (GUI). The classification system extracts and saves the rules held in the classifier (in a text file) in a human-readable form.

Within the prototype already implemented classification algorithms and methods are used; basic learning algorithms are called from *Weka* software (Hall *et al.*, 2009), multi-label classification methods which make use of them are implemented in *Mulan* (Tsoumakas *et al.*, 2011) library. A prototype in the exploitation mode currently uses 11 static learning algorithms or method-algorithm combinations from *Weka* and *Mulan*, applying their default settings.

To implement an interactivity scheme, the classifier's application stage has been improved with the ability to trace the confidence of classification and intercept uncertain classifications. Classification results are presented to the user (expert), which can apprise classes assigned with different confidences and make his classification if no classification is given with the confidence 0.5 or more.

To emphasize the novelty of development differences and improvements in comparison to *Weka* tool and *Mulan* library are summarized. From this aspect the main InClaS contributions are:

(1) The developed GUI for *Mulan* library (developers of *Mulan* do not provide GUI).
(2) The ability for a system's user to examine the classifier rule base conveniently (if a particular learning algorithm produces rules).
(3) GUI and processing engine behind it for ensuring interactivity.

Thus all together the InClaS prototype provides a unique environment for multi-label classification in a more user-friendly way than it was possible before as well as novel interactivity facilities between the classification system and its user.

5. The Application of Inclas in Education

This section describes the experimental plan and main results in practical evaluation of the InClaS model and its prototype in the domain of higher education, the university study course comparison in particular. The aim of experiments is to examine the utility of the InClaS framework, usability of the system's prototype and evaluate the impact of chosen settings to study course comparison task.

In order to assess an InClaS utility the number of misclassified instances, applying the standard non-interactive approach and the proposed interactive approach is to be compared. Regarding usefulness of the proposed solution in education area the following aspects are to be evaluated:

- Verification of the thesis that this problem domain is not appropriate for traditional automatic machine learning solutions, whereas inductive learning methods based interactive multi-label classification system for supporting study course comparison can provide acceptable solution.
- Evaluation of a direct (using attributes achieved directly from full course descriptions) and indirect (using mediated attributes from course descriptions) study course comparison.

5.1. *Experimental Plan*

Experimental settings are described in Table 1.

Four setting combinations as separate *stages of experiments* are defined: (1) word vectors with automatic classification, (2) mediated attributes with automatic classification, (3) word vectors with InClaS, and (4) mediated attributes with InClaS. Stage 1 is preliminary to stage 3 and stage 2 precedes stage 4.

Parameters of data sets are given in Table 2.

The full data set consists of 79 examples from different European universities providing Business Informatics related curricula, namely, 25 instances from Riga Technical University, 6 instances from University of Rostock, 31 from Vienna University of Technology and 17 from University of Vienna. In a reduced set, the labels with less than 4 examples are removed. Label density of a data set is the average number of labels of the examples divided by number of labels. Label cardinality of a data set is the average number of labels of the examples in this set. Distinct labelsets present the number of different label combinations within a data set. Word vector based data set contains 1884 attributes representing appearance or absence of 1884 words encountered in study course description examples. Competency based data set contains 36 attributes representing e-CF competencies, one attribute describing study level and one attribute – number of ECTS credit points.

Table 1

Experimental settings for study course comparison

	Stage 1	Stage 2	Stage 3	Stage 4
Input data set	Full study course descriptions (ex-racting word vectors in preprocessing)	Competencies of study course (e-CF), number of credit points, study level	Full study course descriptions (extracting word vectors in preprocessing)	Competencies of study course (e-CF), number of credit points, study level
Classification approach	Automatic classification		Interactive classification (InClaS)	
Classification algorithms (methods)	20 classification algorithm-method combinations (from *Weka* and *Mulan*)		4 best methods from Stage 1	4 best methods from Stage 2
Evaluation measures	Hamming loss, Micro-average precision, Micro-average recall, One-error, Coverage		*M*, *PC*, *FU*, *TU*	

Table 2

Study course data set

	No. of attributes	No. of instances	No. of classes	Label density	Label cardinality	Distinct labelsets
Full data set (word vectors)	1884	79	25	0.0620	1.6203	52
Full data set (competencies)	38	79	25	0.0620	1.6203	52
Reduced data set (competencies)	38	64	12	0.1341	1.6094	36

5.2. *Main Experimental Results*

Table 3 shows 3 times repeated random sub-sampling validation results (in stage 3) of four methods which achieved the best results by means of Hamming loss, Micro-average precision, Micro-average recall, One-error, Coverage in stage 1. *BR* stands for the Binary Relevance method. Classification measures hold the following correlations:

PC + *Misclassified* (*without interactivity*) = 1 (*all classifications in an automatic manner*).
PC + *TU* + *FU* + *Misclassified* (*with interactivity*) = 1 (*all classifications in an interactive manner*).
Misclassified (*without interactivity*) = *TU* + *FU* + *Misclassified* (*with interactivity*).

Results in Table 3 should be interpreted as follows. Using the automatic classification where only partly or completely correct classifications (blue part of the table) and misclassifications (red part of the table) exist, 27% of instances would be *PC* (in case of RAkEL method) and 73% – misclassified. If the interactive approach is used, the number of *PC* remains the same; however, 33% of instances from previously misclassified are marked as uncertain to the classifier and given to the expert, reducing the number of misclassified instances to 40%. Results in Table 3 show that without applying interactivity the number of misclassified instances is much higher for all methods. Note the assumption that the expert makes correct classifications to the instances passed to him.

Table 4 represents results of stage 4 experiments.

Table 3

Interactive approach for direct study course comparison (word vectors)

Method (algorithm)	Partly correct (*PC*)	True uncertain classification (*TU*)	False uncertain classification (*FU*)	Misclassified (with interactivity)	Misclassified (without interactivity)
RAkEL(*J48*)	0.267	0.333	0.000	0.400	0.733
BR(*AdaBoost*)	0.100	0.400	0.000	0.500	0.900
BR(*Bagging*)	0.067	0.600	0.000	0.333	0.933
BR(*JRip*)	0.267	0.367	0.000	0.366	0,733

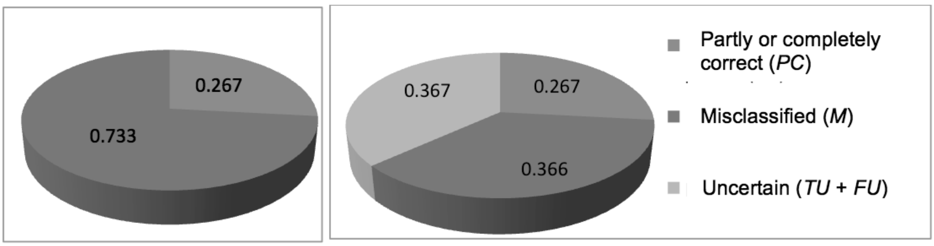
Table 4

Interactive approach for indirect study course comparison (competencies)

Method (algorithm)	Partly correct (*PC*)	True uncertain classification (*TU*)	False uncertain classification (*FU*)	Misclassified (with interactivity)	Misclassified (without interactivity)
BR(*NB*)	0.234	0.633	0.000	0.133	0.766
BR(*Bagging*)	0.167	0.733	0.000	0.100	0.833
BR(*AdaBoost*)	0.267	0.433	0.000	0.300	0.733
BR(*JRip*)	0.267	0.367	0.000	0.366	0,733

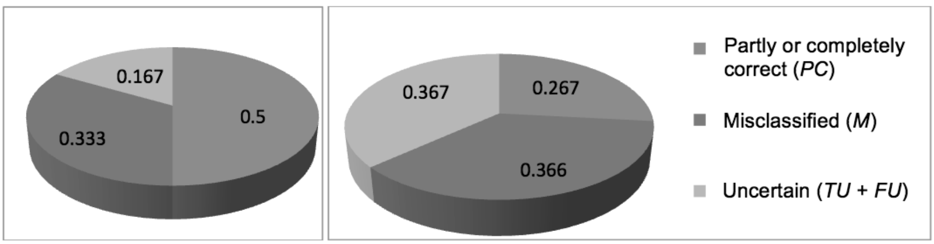
Alike stage 3 results, the ability of the InClaS classification system to track uncertain classifications allows to decrease the number of misclassified instances, although results vary much between the methods used. Graphical representation of JRip algorithm results in Fig. 7 emphasizes the impact of the interactive approach even more. Without interactivity (Fig. 7 part A), all instances in the red column of the table would be misclassified reaching only 27% of *PC*. Such classification results do not encourage the use of the automatic classification in this problem domain. In turn, the interactive approach (Fig. 7 part B) with the ability to handle uncertain classification makes it possible to save half of misclassified instances and assign to them correct classifications after the expert's review. Thus, 37% of instances are misclassified, which, obviously, is not a great result, but is much more promising than 73% with the automatic classification.

To all appearances, the given data set does not provide a complete concept description as it was assumed when considering domain features. To consider the situation when the number of training examples regarding each class has increased, experiments with the reduced data set are carried out. The results lead to conclusion that interactive classification system improves its results and less frequently disturbs the expert when the training set grows in time. Therefore it is useful to spend expert's time more in the initial period of classifier's usage in order to obtain better classification results later. Fig. 8 shows the difference between results in the data set with reduced number of classes where each class is described with slightly higher number of examples (part A) and the full data set which includes many underrepresented classes (part B). In reduced data set *PC* reach 50% of instances leaving 17% of instances for expert's decision and also decreasing the number of misclassified instances. All these parameters are improved in comparison to the initial data set.



A B

Fig. 7. Test results of *JRip* algorithm with automatic (A) and interactive (B) classification.



A B

Fig. 8. Test results of *JRip* algorithm task with reduced (A) and full (B) course data set.

Experimental results also deny assumption that the indirect course comparison provides better classification results than the direct comparison. That is, structured and meaningful information extraction from course descriptions produce attributes which do not surpass full course description usage to make word vector based attributes by means of number of misclassified instances and (partly) correct classifications. Both approaches can be used, however, the indirect comparison currently requires much more expert's work in attribute extraction phase since competencies are not accessible directly in course descriptions. If course descriptions are standardized, it makes the situation more convenient for such approach. As a disadvantage of word vector usage to define attributes its low semantic meaning should be mentioned. It does not provide useful knowledge to the expert as it only describes occurrences of different words in descriptions wherever in the text they appear – either preconditions or learning outcomes. Therefore, the knowledge about underlying communalities of the course content can be mined if meaningful attributes are used, like competencies which the study course provides.

As example of rules generated by competencies-based classifier Fig. 9 shows a section of the JRip classifier which is highly understandable for a human.

Each rule describes one study course based on comparison data set available for classifier training. Therefore, classification model of *Riga Technical University* course *Enterprise Architecture and Requirements Engineering* says that if other course provides competency *Solution Development* (competency B.4 regarding e-CF) than the courses are similar, otherwise they are not. Confidence for these rules are 73% (true for 8 instances, wrong for 3 in training data set) and 90%, respectively.

This type of representation provides expert with easy to evaluate knowledge discovered directly from historical or on-demand created course comparisons between different educational institutions. Corresponding study courses are gathered by examining all models therefore one course can achieve more than one classification.

```
Model for Enterprise Architecture and Requirements Engineering JRip rules:
==========

 (B.4. Solution Deployment = 1) =>
EnterpriseArchitectureAndRequirementsEngineering=1 (8.0/3.0)
 => EnterpriseArchitectureAndRequirementsEngineering=0 (61.0/7.0)

Number of Rules: 2

Model for Quality Risk and Security Technologies JRip rules:
==========

(E.3. Risk Management = 1) and (E.2. Project and Portfolio Management = 0) =>
QualityRiskAndSecurityTechnologies=1 (6.0/1.0)
 => QualityRiskAndSecurityTechnologies=0 (63.0/1.0)

Number of Rules: 2
```

Fig. 9. Excerpt of classification rules for study course comparison competencies-based data set.

6. Conclusions

The analysis of the existing situation on automation of course description correspondence detection has identified that this task is distinctive and do not fit to traditional automatic solutions because of small available data set, semi-structured data sources and multi-label class membership.

Having analysed computer supported educational document comparison and current interactive classification approaches regarding dealing with unclassified instances, the authors of this paper suggest InClaS framework, on which bases algorithms, methods and other components are defined and which allow to develop an interactive classification system for decreasing misclassified instances in domains where a human-expert.

A prototype of an interactive multi-label classification system is developed which is adjusted for study course comparison task. Course correspondences between *Business Informatics* master study programme in *Riga Technical University* and courses of several corresponding study programmes in Europe are detected. Evaluation of the InClaS has been carried out which proved the ability to decrease the number of misclassified instances significantly if uncertain classifications are detected and passed to the expert's review.

However, we can broaden proposed application areas of InClaS framework and do not stick only to educational domain. The following recommendations of InClaS application are drawn.

The use of the interactive classification system is *feasible* in areas where:

- Human-expert is available that can classify individual instances.
- Problem domain is defined by the attributes which are comprehensible for the expert – not too overwhelming in amount and available in a human interpretable form.

The interactive classification approach is *more appropriate* than the automatic classification in areas where at least one of the following statements holds:

- It is essential to receive a correct classification for as much instances as possible, and it is acceptable to invest the expert's work and time to achieve it.
- It is hard to extract or define domain features resulting in attributes which do not describe the underlying concept completely.
- Only a small initial learning set is available and it is suspected not being representable.

The theoretical and practical results provide opportunities for further research. Some of future investigation directions are defining more sophisticated similarity measures and considering other supervised and semi-supervised machine learning approaches for the comparative analysis of university study courses, e.g., co-training and case-based reasoning.

7. Acknowledgements

This work has been supported by the European Social Fund project Nr. 2013/0012/1DP/1.1.1.2.0/13/APIA/VIAA/051.

Bibliography

Aksoy, M.S. (2008). Applications of Rules-3 induction system. In: *Proceedings of the Innovative Production Machines and Systems*.

Alves, H., Figueira, Á. (2011). A educational library based on clusters of semantic proximity. In: *IADIS European Conference on Data Mining*. 226–228.

Anohina-Naumeca, A., Graudiņa, V., Grundspeņķis, J. (2012). Curricula comparison using concept maps and ontologies. In: *Proceedings of the 5th International Scientific Conference on Applied Information and Communication Technology.* Latvia, Jelgava.

Bielawski, L., Lewand, R. (1991). *Intellignet Systems Design: Integrating Expert Systems, Hypermedia, and Database Technologies*. John Wiley & Sons.

Biletska, O., Biletskiy, Y., Li, H., Vovk, R. (2010). A semantic approach to expert system for e-assessment of credentials and competencies. *Expert Systems with Applications*, 37**,** 7003–7014.

Biletskiy, Y., Brown, J. A., Ranganathan, G. (2009). Information extraction from Syllabi for academic e-advising. *Expert Systems with Applications*, 36**,** 4508–4516.

Birzniece, I. (2010). Interactive inductive learning system. In: Barzdins, J., Kirikova, M. (Eds.), *Selected Papers from the DB&IS 2010*. IOS Press, Latvia, Riga, 380–393

Birzniece, I. (2011). Interactive inductive learning based classification system. In: *Proceedings of the IADIS International Conference Intelligent Systems and Agents.* IADIS.

Birzniece, I., Kirikova, M. (2011). Interactive inductive learning: application in domain of education. *Scientific Journal of Riga Technical University, Computer Sciences, Applied Computer Systems*, 47**,** 57–64.

Birzniece, I., Rudzajs, P. (2011). Machine learning based study course comparison. In: *Proceedings of the IADIS International Conference on Intelligent Systems and Agents 2011*. IADIS, 107–111.

Birzniece, I (2013). *Development of Interactive Inductive Learning Based Classification System's Model*, *Doctoral Dissertation*. Riga Technical University.

Brian, R. G., Compton, P. (1995). Induction of Ripple-Down Rules applied to modelling large databases. *Journal of Intelligent Information Systems*, 5**,** 211–228.

Buntine, W., Stirling, D. (1991). Interactive induction. In: Hayes, J. E., Michie, D., Tyugu, Ė. (Eds.), *Machine Intelligence: Towards an Automated Logic of Human Thought.* Clarendon Press, New York.

Clark, P., Niblett, T. (1989). The Cn2 induction algorithm. *Machine Learning Journal***,** 261–283.

Coletta, R., Castanier, E., Valduriez, P., Bellahsene, Z., Duyhoa, N. (2011). WebSmatch: a platform for data and metadata integration. In: *DataRing Project meeting June 20 2011 Montpellier*. Montpellier.

Dagienė, V., Jevsikova, T., Kubilinskienė, S. (2013). An integration of methodological resources into learning object metadata repository. *Informatica*, 24(1), 13–34.

European E-Competence Framework [Online]. (2012). `http://www.ecompetences.eu/`

Hadjimichael, M., Wasilevska, A. (1993). Interactive inductive learning. *International Journal of Man-Machine Studies***,** 147–167.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H. (2009). The Weka data mining software: an update. *SIGKDD Explorations*, 11**,** 10–18.

Jianwu, Y., Chen, X. (2002). A semi-structured document model for text mining. *Journal of Computer Science and Technology*, 17**,** 603–610.

Kennedy, D., Hyland, Á., Ryan, N. (2009). Writing and using learning outcomes: a practical guide. *EUA Bologna Handbook: Making Bologna Work.*

Li, X., Feng, L., Zhou, L., Shi, Y. (2009). Learning in an ambient intelligent world: enabling technologies and practices. *IEEE Transactions on Knowledge and Data Engineering*, 21**,** 910–924.

Michalski, R.S., Mozetic, I., Hong, J., Lavrac, N. (1986). The multipurpose incremental learning system AQ15 and its testing application to three medical domains. In: *5th National Conference on Artificial Intelligence.* Morgan-Kaufmann, San Francisco.

Müller, N. (2010). *Change Management on Semi-Structured Documents.* Jacobs University Bremen

Northern Kentucky University [Online]. (2012). `http://www.nku.edu/`

Okabe, M., Yamada, S. (2002). Interactive web page retrieval. *Active Mining: New Directions of Data Mining.* OS Press, Amsterdam.

Ranganthan, G.R., Biletskiy, Y., Macisaac, D. (2006). Machine learning for classifying learning objects. *IEEE CCECE/CCGEI***,** 280–283.

Riga Technical University [Online]. (2013). `http://www.rtu.lv/`

Rudzājs, P., Kirikova, M. (2009). It knowledge requirements identification in organizational networks: cooperation between industrial organizations and universities. In: *Proceedings of the 18th International Confer-*

ence on Information Systems Development. China, Nanchang.

Rudzājs, P., Kirikova, M. (2012). Towards monitoring correspondence between education demand and offer. In: *21st International Conference on Information Systems Development.* Italy, Prato.

Saleem, K., Bellahsene, Z., Hunt, E. (2008). Porsche: performance oriented schema mediation. *Information Systems*, 33, 637–657.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, 34**,** 1–47.

Settles, B. (2010). Active learning literature survey. In: *Computer Sciences Technical Report 1648.* University of Wisconsin–Madison.

Tanumara, R.C., Xie, M., Au, C.K. (2007). Learning human-like color categorization through interaction. *International Journal of Computational Intelligence*, 4, 338–345.

Teodosiev, T., Nachev, A. (2012). Some pitfalls in introductory programming courses. *Informatics in Education*, 11(2), 241–255.

Thabtah, F.A., Cowling, P., Peng, Y. (2005). Multiple labels associative classification. *Knowledge and Information Systems*, 9, 109–129.

Tsoumakas, G., Katakis, I. (2007). Multi-label classification: an overview. *International Journal of Data Warehousing and Mining*, 3, 1–13.

Tsoumakas, G., Spyromitros-Xioufis, E., Vilcek, J., Vlahavas, I. (2011). Mulan: a Java library for Multi-label learning. *Journal of Machine Learning Research*, 12, 2411–2414.

Wong, M.L., Laung, K.S. (2000). *Data Mining Using Grammar-Based Genetic Programming and Applications*. Kluwer Academic Publishers, USA.

I. Birzniece, Dr.sc.ing., is a researcher in computer science field and director of Study Department at Riga Technical University, Latvia. She has been participating in international scientific society by means of publications, scientific projects and conference organization committees and now takes part in ESF project "Establishment of interdisciplinary ICT research group for transmission, processing and managing for a large volume of data". Research interests: machine learning, data mining and educational data analysis.

P. Rudzajs, Mg.sc.ing, is a PhD student at Faculty of Computer Science and Information Technology, Institute of Applied Computer Systems, Riga Technical University, Latvia. He works as a researcher at Faculty of Computer Science and Information Technology in Institute of Applied Computer Systems and as a head of Information Systems Unit in Information Technology department, Riga Technical University. He is author and co-author of more than 20 papers in the field of information systems development. Research interests: information service systems development and conceptual modelling, information retrieval and extraction.

D. Kalibatiene, assoc. prof. dr., is full time docent at the Department of Information Systems, and researcher at the Information Systems Research Laboratory in Vilnius Gediminas Technical University. She participated in the High Technology Development Program Project "Business Rules Solutions for Information Systems Development (VeTIS)". She is the member of the European Committee and Lithuanian Government supported SOCRATES/ERASMUS Thematic Network project "Doctoral Education in Computing" (ETN DEC), "Teaching, Research, Innovation in Computing Education" (ETN TRICE) and "Future Education and Training in Computing: How to Support Learning at Anytime Anywhere" (ETN FETCH). She is author and co-author of more than 30 papers and 1 book in the field of information systems development. Research interests: business rules and ontology based information systems development, conceptual modelling and rule based dynamic business process modelling and simulation.

O. Vasilecas, prof. (hp) dr., is full time professor at the Information Systems Department, principal researcher and head of Information Systems Research Laboratory in Vilnius Gediminas Technical University. He is author and co-author of more than 370 research papers and 5 books in the field of information systems development. His research interests include knowledge based information systems, knowledge represented by business rules and ontology, knowledge extraction from information systems databases, dynamic business processes, information systems modernisation. He delivered lectures in 7 European universities including London, Barcelona, Athens and Ljubljana. O. Vasilecas carried out an apprenticeship in Germany, Holland, China, and last time in Latvia and Slovenia universities. He supervised 11 successfully defended doctoral theses and now is supervising 4 more doctoral students. He was leader of number of international and local projects. Last time he leaded "Business Rules Solutions for Information Systems Development (VeTIS)" project carried out under High Technology Development Program of Lithuania.

E. Rencis, Dr.sc.comp., is a researcher in computer science field in Institute of Telecommunications at Riga Technical University, Latvia. He has been participating in international scientific society by means of publications, scientific projects and conference organization committees and now takes part in ESF project "Establishment of interdisciplinary ICT research group for transmission, processing and managing for a large volume of data". Research interests: machine learning, data mining and fiber optical transmission systems.

Interaktyvios klasifikavimo sistemos taikymas universitetinių studijų programoms palyginti

Ilze BIRZNIECE, Peteris RUDZAJS, Diana KALIBATIENE,
Olegas VASILECAS, Edgars RENCIS

Augant informacijos kiekiui atsirado poreikis ją klasifikuoti pagal apibrėžtus kriterijus. Ši klasifikacijos problema yra aktuali ir aukštojo mokslo srityje, ieškant panašių studijų programų ir studijų modulių, kas suteiktų galimybę įgyvendinti studentų mainus tarp universitetų ir palengvintų studijų modulių administravimą. Šiuo metu studijų modulių palyginimas ir administravimas yra rankinis darbas, kurį galima būtų automatizuoti įdiegus intelektualiąsias bei adaptyviąsias sistemas. Šios probleminės srities duomenys dažnai yra nestruktūrizuoti, pateikti teksto pavidalu. Tai apsunkina klasifikavimą, o egzistuojantys tokiems uždaviniams spręsti algoritmai nepakankamai palengvina darbą. Straipsnio autoriai siūlo klasifikavimo sprendinį, kuris leidžia iš dalies automatizuoti klasifikavimo procesą, įtraukus ne tik dalykinės srities ekspertus, bet ir intelektualias sistemas. Remiantis pasiūlytu sprendiniu sukurtas jį realizuojantis prototipas ir atlikti bandymai, kurie parodė siūlomo metodo veiksmingumą.