Krassimir Markov, Vladimir Ryazanov,
Vitalii Velychko, Levon Aslanyan
(editors)

New Trends in Classification and Data Mining

I T H E A
SOFIA
2010

Krassimir Markov, Vladimir Ryazanov, Vitalii Velychko, Levon Aslanyan (ed.)

New Trends in Classification and Data Mining

ITHEA®

Sofia, Bulgaria, 2010

First edition

Recommended for publication by The Scientific Concil of the Institute of Information Theories and Applications FOI ITHEA

This book maintains articles on actual problems of classification, data mining and forecasting as well as natural language processing:

- new approaches, models, algorithms and methods for classification, forecasting and clusterisation. Classification of non complete and noise data;
- discrete optimization in logic recognition algorithms construction, complexity, asymptotically optimal algorithms, mixed-integer problem of minimization of empirical risk, multi-objective linear integer programming problems;
- questions of complexity of some discrete optimization tasks and corresponding tasks of data analysis and pattern recognition;
- the algebraic approach for pattern recognition - problems of correct classification algorithms construction, logical correctors and resolvability of challenges of classification, construction of optimum algebraic correctors over sets of algorithms of computation of estimations, conditions of correct algorithms existence;
- regressions, restoring of dependences according to training sampling, parametrical approach for piecewise linear dependences restoration, and nonparametric regressions based on collective solution on set of tasks of recognition;
- multi-agent systems in knowledge discovery, collective evolutionary systems, advantages and disadvantages of synthetic data mining methods, intelligent search agent model realizing information extraction on ontological model of data mining methods;
- methods of search of logic regularities sets of classes and extraction of optimal subsets, construction of convex combination of associated predictors that minimizes mean error;
- algorithmic constructions in a model of recognizing the nearest neighbors in binary data sets, discrete isoperimetry problem solutions, logic-combinatorial scheme in high-throughput gene expression data;
- researches in area of neural network classifiers, and applications in finance field;
- text mining, automatic classification of scientific papers, information extraction from natural language texts, semantic text analysis, natural language processing.

It is represented that book articles will be interesting as experts in the field of classifying, data mining and forecasting, and to practical users from medicine, sociology, economy, chemistry, biology, and other areas.

General Sponsor: Consortium FOI Bulgaria (www.foibg.com).

Printed in Bulgaria

ISBN 978-954-16-0042-9

C\o Jusautor, Sofia, 2010

DATA AND METADATA EXCHANGE REPOSITORY USING AGENTS IMPLEMENTATION

Tetyana Shatovska, Iryna Kamenieva

***Abstract**: For implementation of an intelligent data and metadata exchange repository the intelligent agent oriented approach has been selected. In this work the conceptual structure and interaction principles of intelligent agents and ontological models in the intelligent data and metadata exchange repository will be offered. The main attention in this work will be paid to the development of intelligent search agent model realizing information extraction on ontological model of Data mining (DM) methods. In a client part of system there is considered the building of the intelligent agent of the repository user, the coordinator (manager) agent, which controls the common state of the system, and also fulfils the registration and authorizations of users, the resource (dataset) agent with partial usage of files structure with SDMX standard data. The model uses the service oriented architecture. Here is used the cross platform programming language Java, multi-agent platform Jadex, database server Oracle Spatial 10g, and also the development environment for ontological models - Protégé Version 3.4.*

***Keywords**: repository, SDMX standart, data mining, semantic web, ontology, multiagent system, search algorithms, agent-oriented systems, intelligent agent, jadex, sdk, java, rdf, protégé, sparql, oracle splatiat.*

***ACM Classification Keywords**: H.3.3 Information Search and Retrieval*

Introduction

Digital repositories are networked software applications primarily used for storing, managing and disseminating data (e.g. digital publications, theses, data sets and so on). The Repositories differ from conventional content management systems because they include technologies to ensure that data are preserved for long-term access and use.

We are focused on developing multi-agent system for processing and storage of any statistical data. Research of existing repositories allowed identifying the main bottlenecks of the similar statistical repositories that were taken into account. Operating with UCI repository the user is able to filter, according to subsection of data mining area, the data files to view the brief characteristic of a file, to download a file. Using DEA Dataset Repository the user is able to search in any of criteria, view the brief characteristic of a file and to download a file, but only after .XML registration. In Data Repository the user can download any file of subjects without registration. Operating with Frequent Itemset Mining Dataset Repository the user does not need to be registered, he can obtain the information about researches made on samplings and the contact information of researchers, to download a file.

A key feature of the developed system via above mentioned typical statistical repositories is implementation of the datasets metadescription using the European standard SDMX 2.0 and ontological models that are stored in the system.

The advantage and novelty of the work is implementation of an ontological models set of the Data mining methods, which is used for the selection of a proper method under the sample source of the user datasets. To work with set of the ontological models have been developed a set of search algorithms that implement simple and advanced search supporting, account search, which takes individual user interests, direction of scientific activities, previous search queries of the user, as well as architecture of search module based on these search algorithms. Also our system (intelligent data and metadata exchange repository) has a taxonomy of DM

methods that allows to establish connection between DM methods and problem domain data on which they could be applied. The user ontological model, resources ontological model have been developed in protégé version 3.4.

For ontological models interaction and implementation of search algorithms it was developed a set of general intelligent agents models. They can be used as a mechanism for displaying information on the ontological models, as well as a mechanism for user interaction with the system. This set of general models include model for integrating intelligent agents with web systems, a model of intelligent search agent, and model for relationship between agents. The user of the developed intelligent data and metadata exchange repository is able to make formal description of the user's problem domain (filling in the necessary fields in the ontology model) and formal description of the dataset which is need for specific tasks. All this kind of activities is a part of the search agent. The search agent, having processed the received information, transfers it to the coordinator agent and via the search agent the necessary connection with a data file is made. The user intelligent agent (user agent (profile agent)) allows to personalize the answer to the following questions: what is the user name; what is e-mail address; what language the user prefers; what are current goals of the user; whether the user is beginner or advanced one; what academic institution the user belongs to; what are localization preference of the user; whether the interests of user coincide with other users interests in the system; what are recent inquiries of the user. The result of applying the multi-agent approach for creating such system is the ability to perform a simple search for users regardless of user type; to search by different criteria for authorized users; to provide popular data sets; to perform a search taking into account the personal needs of the user; to provide user relevant queries information; to keep statistics of requests and, if necessary, provide this information; to remember the successful search results.

Here is used the cross platform programming language Java, multi-agent platform Jadex, database server Oracle Spatial 10g, and also the development environment for ontological models – Protégé Version 3.4. Database management system Oracle Spatial 10g which allows to work with ontologies in RDF format was chosen as a method of resource ontological models storage. Development environment of ontological models is Protégé.

Intelligent search agent design and development

As one of basic concept of DMDR system is search agent.

For searching in the data and metadata exchange repository we have to develop a search module. It would consider the current state of system and different searching criteria to adopt any strategy of search [Ratushin, 2001]. One of the most suitable solutions to this problem is the intelligent agents based on goal. This intelligent agent will act not just in reflective way when a request came, but would decide what actions are needed to achieve its goals in terms of the current state of environment. This agent is not able to supervise the environment, where it's executed, in full [Wooldridge, 1995].

In the search module of "data and metadata exchange repository" is set problems such as simple and advanced search or personal search. On the other hand, the search agent is used in the multi-agent environment and agent needs to communicate between it-self and other agents as well as to exert the medium, where it's executed. From these two points of view the functionality of search agent may be divided into functionality in terms of user and functionality in terms of other agents and execution environment [Russell, 2006].

Functionality in terms of user should include the following basic set: to execute a simple search only for non-authorized user, to execute the various searches for authorized user and to provide useful services for the search (Figure 1).

Both authorized and non-authorized users may execute the simple search, in both cases there will be shown the most popular data sets in the repository or the most popular queries (queries most often made by users in the

repository) and the results may also be hints as content search queries (queries correlated with the current ones). But still the authorized user has more privileges in comparison with non-authorized one. The following is available for authorized user: advanced search (search by various data set criteria). When agent uses the search there is displayed some of recent queries. Information about user's requests and their results are stored using a personal agent and will be used further to provide user with more relevant results considering his previous requests.

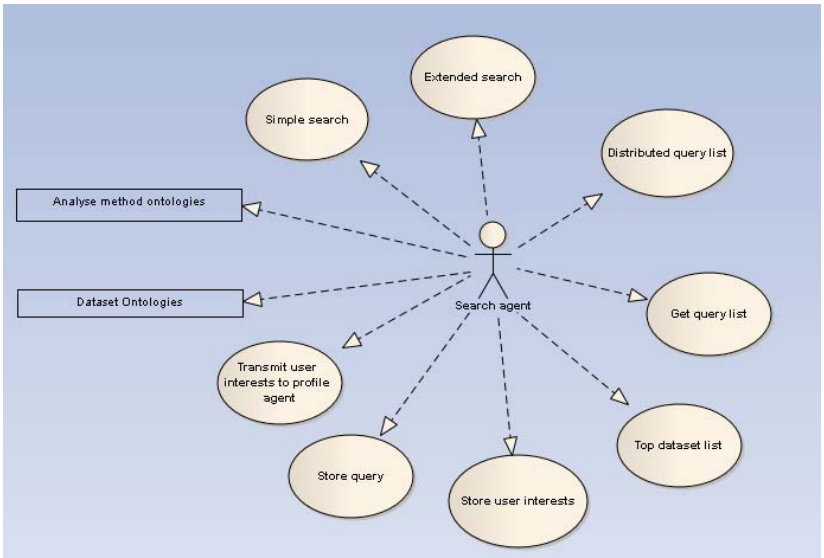

Figure 1 The use case diagram in terms of user

On the other hand, the search agent must interact with other agents to successfully achieve the goals. To render the useful information to them and to request the necessary information from them or to request them to provide a service. Personal Agent of user obtains the results from the search agent and the request itself, as well as to take the transition sequence of user until the user will find the necessary dataset. This information will be stored by the private agent in order that the search agent could further use it.

User agent Scenario

The base information unit of the personal agent is ontology model of user . At the level of agent conception about the user is object model of user ontology. The main objective of the personal agent is to transfer information about users to other agents and to transfer the necessary information to user from the other system agents [Zaborovski, 2005]. So, personal agent should be able to form answers to queries from other agents of "data and metadata exchange repository" system and to modify the user profile during his work with the system. In accordance with the information and ontology model of user the personal agent should be able to form answers to questions related to user. We can allocate the following two partitions of information about user: personal information about user, information about current goals of user. In general case the personal agent should be able to respond the following questions: what is user name, what is his e-mail address, what language user prefers, what are the current goals of user; is user advanced or beginner (naïve, simple), what academic institutions the user belongs to; what are localization preference of the user; are the interests of user coincide with other users interests in the system; what are the recent requests of the user. Here the personal agent applies the developed information and ontology model of the user for questions, which can be requested by other agents while interaction with personal program agent during the work of user with the system [Xacken, 2005].

The User Agent is created after the user authentication. Figure 2 shows the User Agent functionality. When authentication and authorization are completed the user agent retrieves its knowledge information about users that later allows the user and other agents to access this information quickly. The user agent stores this information in its knowledge when active user is in the system and before the work cessation; the agent unloads this information into the database. The User Agents in the system as much as users have passed authentication at the current period. If the user does not operate with agent over 30 minutes, the user agent removes the search agent and itself from the system.

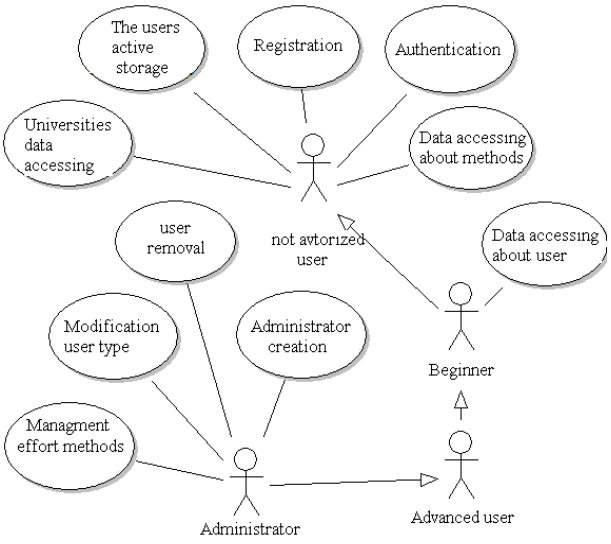

Figure 2 Use case for user agent

The user sets the following tasks before the user-agent:

- user personal data storing;
- changing of user data;
- preserve a user's search query to the system;
- conservation of user activity in the system;
- tracking the status of the user in the system;
- retention of the data sets loaded into the system;
- retention of the data sets unloaded from the system;
- User communication with other users of the system.

Manager agent Scenario

To manage the overall system, registration and authorization of users in a “data and metadata exchange repository” operates the manager agent [Zimmermann, 2006]. The manager agent always suspends user and other agent’s queries. Agent Manager exists in the system as a single copy. Agent Manager is parallelized by agent platform. Manager Agent provides functionality from the standpoint of the user schematically shown in Figure 3.

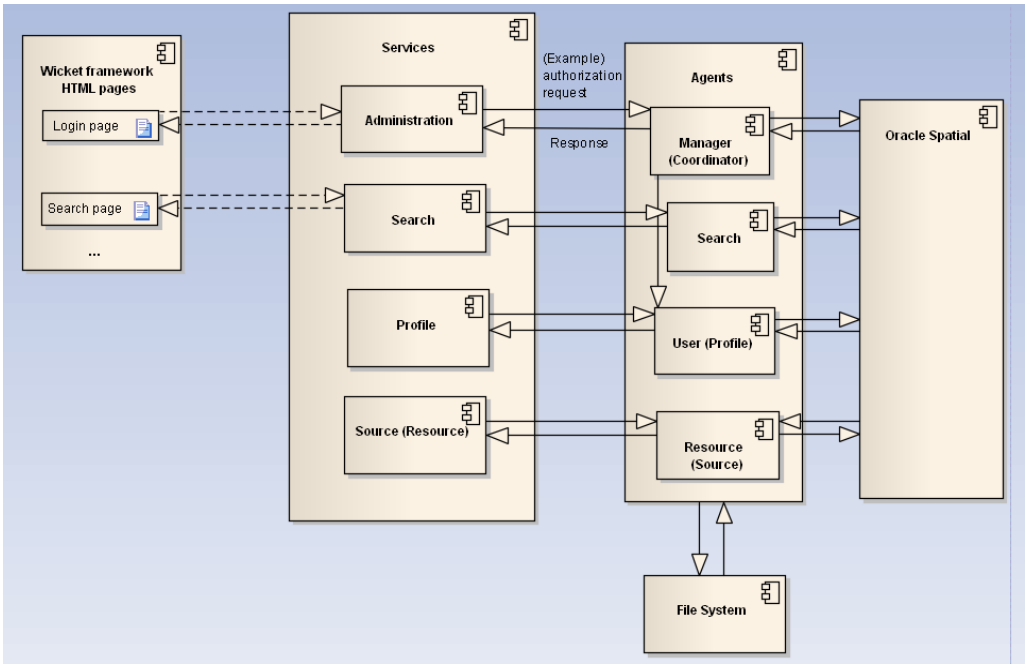

Figure 3 The explanation of overall system

Manager Agent stores the following internal information about the current state of the system:

- the number of users, who use the system;
- the number of beginners, who use the system;
- the number of advanced users, who use the system;
- research methods;
- general information about the universities of the system.
- Manager Agent receives information from the end users and from the environment. Input information from end-user of the system:
- account and password;
- user registration data;
- user account that needs to be transferred into the status of the administrator;
- user account that to be deleted.

End users interact with the agent manager. They invoke a Web service manager from the user interface.

System Administrators remove the users from the system by sending a request to the manager agent. For user to obtain the administrator rights, the other user-administrator should send the request to create a new administrator account on the basis of an existing one. Input information from the user agent: user account whose status should be changed. The transition from the one status to another is carried out by manager agent at the request of user agent of specific user. The Algorithm for the transition as follows: the user agent monitors the user activity in the system, and after getting some experience in the system, the agent prompts the user to raise his status and to receive additional options. If the user agrees the agent sends a request to the manager agent to change the type of user. Also the user invites to enter additional information about himself to obtain additional options. Manager

Agent sends the information to other agents and transmits the information to the end user through a Web service [Fatudimu, 2008].

Source agent Scenario

The main functions of the source agent are:

- scientific data sets addition;
- interaction with the user agent to display the newly added samples to the user depending on the user's interests. The Source agent informs the user agent about adding of scientific datasets to show users information about it after adding a new set to the storage;
- Metadata of datasets edit. The users, who create system or administrator have the possibility to edit ;
- metadata dataset extract from repository;
- selection of entire information about a specific dataset and detailed information may be viewed only by registered users;
- to establish the dataset status numbers of downloads depending on estimates. The rating can be mark to each dataset. The rating assigned using the professional coefficient of a user, who makes it. At the moment of assess its assessment multiplied by a coefficient. This function performs source agent. The source agent should request the user agent ratio, calculate the result and save it in the database. Status of sampling can also increase depending on the number of downloads;
- interaction with the user agent to modify the coefficient of user professionalism depending on the status of scientific data sets, which he has added to the assessment or in storage;
- datasets filtering of metadata datasets by a specific parameter;
- new datasets adding to the repository that were found by search agent in the Internet.

The system must know the properties of the agent to create and run the agent. The state of the agent is determined by beliefs, goals, current plans, as well as libraries of known plans. Jadex uses the declarative and procedural approaches for implementing the components of the agent. The body of the plan is executed as ordinary Java classes. All other notions (beliefs, goals, filters, and conditions) are defined by language. They are allowed to create Jadex objects in a declarative manner. The program developer can refer to the Java code, for example, to define methods. Full identification of the agent is reflected in the so-called agent definition file (ADF). In the ADF file the developer defines the initial beliefs and goals, announcing Java facilities. Announce plans to show the necessary classes from Java code. In addition to the BDI components in ADF file can be stored, some other information, for example, the default arguments for starting the agent or service descriptions for the registration of the agent in the facilitator directory. The structure consists of Jadex API, performed by the model, reusable common features. API provides access to the concept Jadex during programming plans. Plans are obvious classes Java. It is extend a special abstract class which provides a useful method of sending messages, the organization of secondary objectives or expectations of the events. Plans are able to read and modify the agent's thoughts. It uses the API framework agreement. Special function Jadex is that, in addition to the direct extraction of the remaining facts, intuitive OQL - like query language is allow to formulate a random complex expressions using the facilities which are contained in the database views. In addition to plans, coded in Java, provides the developer based on the XML agent definition files (ADF). It establishes the initial thoughts, objectives and plans of the agent. The Jadex mechanism reads file and starts the agent. It tracks its goals during a continuous selection of steps and launches a plan based on internal events and messages from other agents. Jadex is equipped with some advance features - such as access to the directory facilitator service. Feature encoded in the individual plans, linked agent used in many modules which are called abilities. Ability is described in a format similar to the ADF. It can be easily incorporated into existing agents. So summarize, in Jadex agents

is thought, can be any type JAVA-site and stored in the database views. Objectives - explicit or imply descriptions of conditions that must be achieved. The agent executes the plans to achieve their goals. They are JAVA code procedural means.

Currently, there are many repositories of scientific datasets [Bresciani, 2004]. The main disadvantages occurred in these systems are: text-only format is not convenient to use and to change the format of files, not user-friendly interface, and the search is only by one of many criteria, i.e. not allowed to combine the search for a number of conditions, poor search.

In many systems, there is no any understanding for what tasks you can use this dataset, there is also insufficient information on the data. Currently, the agent technologies are widespread, where the main part is the agent - a software entity capable of such qualities as autonomy, activity, commitment, mobility, sociability. The creation of ontologies is a prospective direction of up-to-date research in processing of information provided in natural language. One of the advantages of using ontologies as a tool for learning is a systematic approach to the study of the subject area. Meanwhile achieved: regularity - Ontology provides a holistic view of the subject area, uniformity - the material presented in a unified format is much better perceived and reproduced; scientific - Building the ontology allows to restore the missing logical link in their entirety. Also, ontologies allow the use the great volumes of data from different systems, due to the fact they creating the semantic description of data. a) studied the main stages of work with the repository of scientific research data sets; b) reviewed the existing repositories of scientific data sets, to identify their strengths and weaknesses; c) studied the technology Semantic Web; d) investigated the possibility of agent technology; e) analyzed the ways to develop a web-oriented multi-applications; f) developed the architecture of multi-repository of scientific data sets; g) developed the ontological model of the user; h) developed and realized as a software BDI agent model of the user; i) developed and realized as a software BDI agent model.

Conclusion

The results of the research is developed multi-agent system for processing and storage of any statistical data. A key feature of the developed system via others typical statistical repositories is implementation of the datasets metadescription using the European standard SDMX 2.0 and ontological models that are stored in the system.

The advantage and novelty of the work is implementation a set of the ontological models of Data mining methods, which is used for the selection of a proper method under the sample source of the user datasets. To work with set of the ontological models have been developed a set of search algorithms that implement simple and advanced search supporting, account search, which takes individual interests, orientation of activities, previous search queries of the user, as well as architecture of search module based on these search algorithms. Also this system (intelligent data and metadata exchange repository) has a taxonomy of DM methods that allows to establish connection between DM methods and data on which they can be applied, that for the user of "beginner" class represents itself as the expert system. The user ontological model, resources ontological model have been developed in protégé version 3.4, which allows working fast with ontologies.

For ontological models interaction and implementation of search algorithms it was developed a set of general intelligent agents models. They can be used as a mechanism for displaying information on the ontological models, as well as a mechanism for user interaction with the system. This set of general models include model for integrating intelligent agents with web systems, a model of intelligent search agent, and model for relationship between agents. The user of the developed intelligent data and metadata exchange repository is able to make formal description of the user's problem domain (filling in the necessary fields in the ontology model) and formal description of the dataset which is need for specific tasks. All this kind of activities is a part of the search agent. The search agent, having processed the received information, transfers it to the coordinator agent and via the

search agent the necessary connection with a data file is made. The user intelligent agent (user agent (profile agent)) allows to personalize the answer to the following questions: what is the user name; what is e-mail address; what language the user prefers; what are current goals of the user; whether the user is beginner or advanced one; what academic institution the user belongs to; what are localization preference of the user; whether the interests of user coincide with other users interests in the system; what are recent inquiries of the user. The result of applying the multi-agent approach for creating such system is the ability to perform a simple search for users regardless of user type; to search by different criteria for authorized users; to provide popular data sets; to perform a search taking into account the personal needs of the user; to provide user relevant queries information; to keep statistics of requests and, if necessary, provide this information; to remember the successful search results.

Acknowledgement

The paper is published with financial support by the project ITHEA XXI of the Institute of Information Theories and Applications FOI ITHEA (www.ithea.org) and the Association of Developers and Users of Intelligent Systems ADUIS Ukraine (www.aduis.com.ua).

Bibliography

[Ratushin, 2001] Ratushin U., Polenok, S., Tkachenko, S. Information society ontology at the network. In: University book.

[Wooldridge, 1995] Wooldridge, M., Jennings, N. (1995). Intelligent agents: Theory and practice. In: The Knowledge Engineering Review 10(2), 115-152.

[Russell, 2006] Russell, S., Norvig, P. Russian translation of Artificial Intelligence: A Modern Approach, 2nd Edition, Translated by Ptitsyn K. Ed: Moscow: Williams Publishing, ISBN Press, 356.

[Zaborovski, 2005] Zaborovski, V. Intelligent technologies, 324.

[Xacken, 2005] Xacken, G. Information and self-organization. Macroscopic approach to Complex system, 248.

[Gennari, 2002] Gennari, J. The Evolution of Protégé. An Environment for Knowledge-Based Systems Development.

[Xie, 2006] Xie T., Pei, J. MAPO: mining API usages from open source repositories. In: Proceedings of the International Workshop on Mining Software Repositories (MSR '06), Shanghai, China, ED: ACM Press, New York, 54-57.

[Zimmermann, 2006] Zimmermann, T. Knowledge Collaboration by Mining Software Repositories. In: Saarland University, Saarbrucken, Germany

[Fatudimu, 2008] Fatudimu I.T., Musa, A.G., Ayo, C.K, Sofoluwe, A. B. Knowledge Discovery in Online Repositories: A Text Mining Approach. In: European Journal of Scientific Research, 22 (2), 241-250. ED: EuroJournals Publishing

[Bresciani, 2004] Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., Mylopoulos, J. Tropos: An agent-oriented software development methodology. In: Journal of Autonomous Agents and Multi-Agent Systems 8 (3), 203–236

Authors' Information

Shatovska Tetyana *– Ass.Prof, Kharkiv National University of Radioelectronics, Kharkiv-166, av.Lenina 14, Ukraine; e-mail:shatovska@gmail.com*

Major Fields of Scientific Research: Data and Web mining, Artificial Intelligence

Iruna Kamenieva *–PhD student, Kharkiv National University of Radioelectronics, Kharkiv-166, av.Lenina 14, Ukraine; e-mail: irina.kamenieva@gmail.com*

Major Fields of Scientific Research: Data and Web mining, Artificial Intelligence