ATLANTIS PRESS

Advances in Intelligent Systems Research, volume 174

Proceedings of the 8th Scientific Conference on Information Technologies for Intelligent Decision Making Support (ITIDS 2020)

Generating Data on Individual Learning Paths for Classification Using Deep Learning Networks

Sofia Sosinskaya*
Irkutsk National Research Technical University
Russia, Irkutsk
sosinskaya@mail.ru

Sofia Rogacheva
Irkutsk National Research Technical University
Russia, Irkutsk
pyatsot@bk.ru

Andrey Dorofeev
Irkutsk National Research Technical University
Russia, Irkutsk
dorbaik@istu.edu

Roman Dorofeev
Irkutsk National Research Technical University
Russia, Irkutsk
rdobermann@list.ru

***Abstract*—The article considers information about individual learning paths as a set of observations with a large number of features. To reduce the amount of information, it is proposed to classify samples using deep learning networks. Selections are saved either in a relational database or as an ontology. To device training algorithms, a generation of model data instead of real ones, which is currently missing, is proposed. Data are converted from relational tables to ontology using the ANTLR compiler generator. Based on the generated data and qualimetric estimates, the integral characteristics are calculated using a method also proposed in the article. The final results will be a quality assessment of the classification based on deep learning networks. The methods and approaches proposed by the authors are universal and can be used in any higher educational institution to select an algorithm for their qualitative classification.**

Keywords—*ontology, individual learning paths, deep learning networks, compiler generator, generation of model data*

I. INTRODUCTION

There are several concepts of free education. For example, the educational program of universities offer a set of courses for a specialty, while an individual student is allowed to choose a personal subset among them. Setting up the process of generating test data and optimizing this process for the specifics of the subject area corresponding to the tests. According to authors of the approach based on individual learning paths [1], it is important to introduce students to the special mode of the university educational process at the first year and establish the basis on which they will be able to build their schedule in the future, taking into account that the block of compulsory general education disciplines (core) has enough volume and meets the state educational standards.

At senior courses, the role of special disciplines in the professional sphere (major) increases. An equally important link is electives. Due to the personal choice of preferred subjects, it is possible to form an individual curriculum for each student. The student evaluates the content of the course subjects, his/her preferences based on interests or the expected future career, and selects the subjects to study, thus, forming an individual schedule, a plan, getting an opportunity of deeper understanding the chosen disciplines.

There are many works dedicated to the design of individual learning paths. There are various approaches to assessing the quality of their classification; one of them, which the authors of the article also adhere to, is the use of qualimetric assessments of the disciplines included in the trajectory and the entire trajectory. The issue of assessing the quality of education results, using qualimetric expertise, has been risen in the works of Russian authors O.Yu. Cherednichenko, N.V. Zolotko, T.M. Shamsutdinova

The novelty of the approach proposed by the authors of this paper is in the use of deep learning networks for the classification of trajectory samples. Classes are categories assigned to subsets of values for evaluating the quality of individual trajectories.

However, due to absence of the input data (universities do not collect data on the individual leering paths), we have to use generated input data, which correspond to real ones with university curriculum properties and study subject specifics.

II. CLASSIFICATION OF INDIVIDUAL LEARNING PATHS USING DEEP LEARNING NETWORKS

Individual learning paths accumulated in an university information system are considered as observations received at the input of the deep learning network [2]. Observations X are sets of features ($X_1, X_2,...,X_n$) that supposed to be either numeric or textual. An example of deep learning network architecture with two hidden layers is shown in Fig. 1.

Based on the input features which form the input layer, the output classifying feature is formed in the output layer. The summary information from the output of the previous layer is transmitted to the input of the neurons of the next layer with the weights $W_{i,j}$. After the network is created, the training stage takes place, resulting in the weights being adjusted depending on the training set of observations. After completing the training phase, the network is ready to

classify new observations. We will consider a variant of a network, which is trained supervisory, *i.e.*, the training sequence includes both input features and output classifying features. In this way, a large set of individual paths will be divided into a number of classes defined by the numbers or category names.

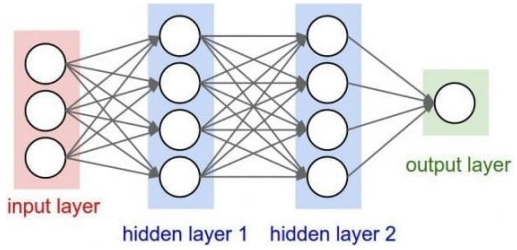

Fig. 1 Deep learning network architecture with an input layer (four input attributes), two hidden layers, and one output in the output layer

The main data source of information about the individual learning paths is a relational database. This method of storing has many advantages, but also a number of disadvantages:

- the result of a query to the database is information stored in the database itself, it is not possible to find data as the result of logical inference;
- using a database, it is difficult to implement complex relationships between objects.

The ontology [3-4] is a modern form of knowledge representation allowing automated processing of information semantics, it enables one to overcome the listed disadvantages of relational databases.

Information stored in an ontology is considered as training samples when using deep learning networks to classify textual information. Text is a sequence of words that have hidden interrelations expressing classifications. To reveal those classifying relations, one can use deep learning of certain types of neural networks .

III. Fundamentals of the Qualimetric Approach to Evaluating Individual Learning Paths

The authors suggest using a qualimetric approach for quality assessments. It is known that the qualimetry is the science of assessment of the quality of an object as a set of features in the form of a computational model [5,6]. This model is a description of features of an object that have certain estimates of their importance (weights). In the problem under consideration, features are the disciplines included in individual learning paths; for each discipline of the module, a weight $p_i \in [0,1]$ is provided, the teacher leading the discipline sets a rating val_i, which is reduced to a qualimetric scale according to the formula:

$$q_i = \frac{val_i - ng}{vg - ng}, \quad (1)$$

where $i=1...n$ is the discipline number,
n is the number of disciplines,
vg is the upper bound of the estimate,
ng is the lower bound of the estimate.

As a result of studying each discipline, the student receives an evaluation c_i.

The individual learning path is provided with an output feature, a class, the integral qualimetric characteristic, calculated by the formula

$$klass = \sum_{i=1}^{n} p_i\, q_i + \sum_{i=1}^{n} \frac{c_{i-1}}{5}. \quad (2)$$

In formula (2), the first term is an integral assessment of the items included in the trajectory, and the second one is an integral assessment of the results of mastering these items. With these characteristics, a transition to classes (categories), which are associated with subsets of values, is carried out for assessing the quality of individual trajectories (terms). Thus, each integral trajectory is represented as

$$X^{(k)} = \left\| Y_{i,j}^{(k)} \right\|, \quad (3)$$

where $i=1...n$ is the discipline number,
n is the number of disciplines,
$j=1...5$ is the discipline characteristics:
$j=1$ is name of the discipline;
$j=2$ is the number of teaching hours;
$j=3$ is an assessment of the discipline by the teacher;
$j=4$ is the percentage of the discipline in the module;
$j=5$ is an assessment of the mastery of the discipline by student.

IV. Structure of the Developed System

The authors propose the structure of the system, shown in Fig. 2.

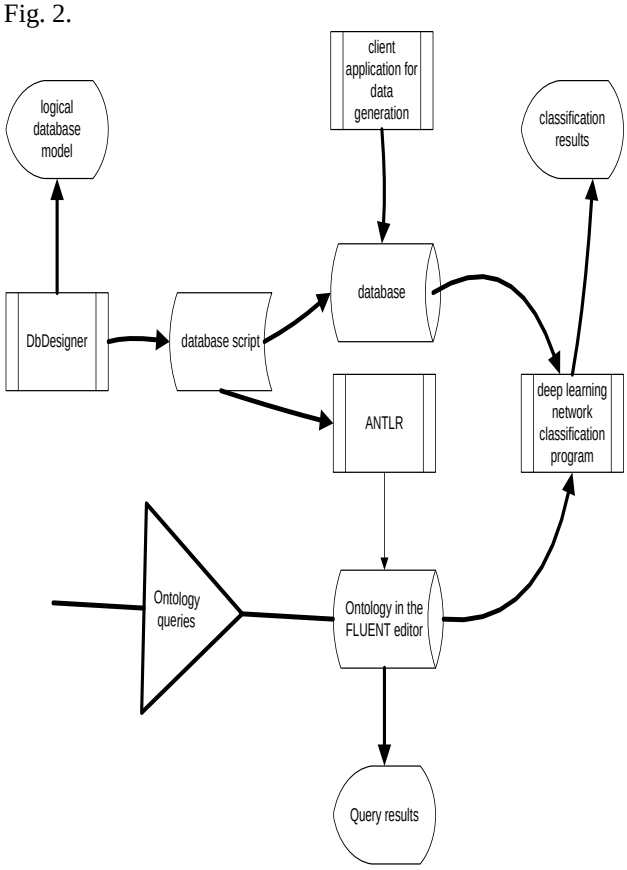

Fig. 2 Structure of the developed system

V. The Structure of the Database

Based on the methodology described above for developing individual learning paths, the main relationships were identified, they are presented in Table 1.

TABLE I. DATABASE RELATIONSHIPS AND THEIR ATTRIBUTES

Relationship name	Relationship attributes
Institute (institute)	Name of the Institute
Specialty (speciality)	Code, name, manager
Specialty of the Institute (ins_spec)	Institute, specialty
Department (kafedra)	Institute, name, head of Department
Teacher of Department (employee)	Full name, position, name of the Department, degree, title, experience, age
Course of the Institute's specialty (course)	Specialty, course number
Course module (module)	Course, module type
Discipline of the Department module (discipline)	Module, Department, name, weight of the discipline
Course group (grp)	Course, group code
Group student (student)	Group, full name, date of birth
Discipline teachers of the Department (disc_emp)	Teacher, discipline, number of lectures, number of laboratory works, number of practical works, availability of course project / work, type of final certification (exam / test), qualimetric evaluation
Subject studied by students (disc_stud)	Student, discipline, received certification assessment
Individual learning path of student (ind_stud)	Student, individual learning path discipline, category, paths

As a result of identifying relationships, designing and normalizing the database to eliminate the redundancy by decomposing them, a logical data model has been designed using DbDesigner shown in Fig. 3. According to this model, the database TRAEKT has been installed on the MS SQL server. A list of its tables is shown in Fig. 4.

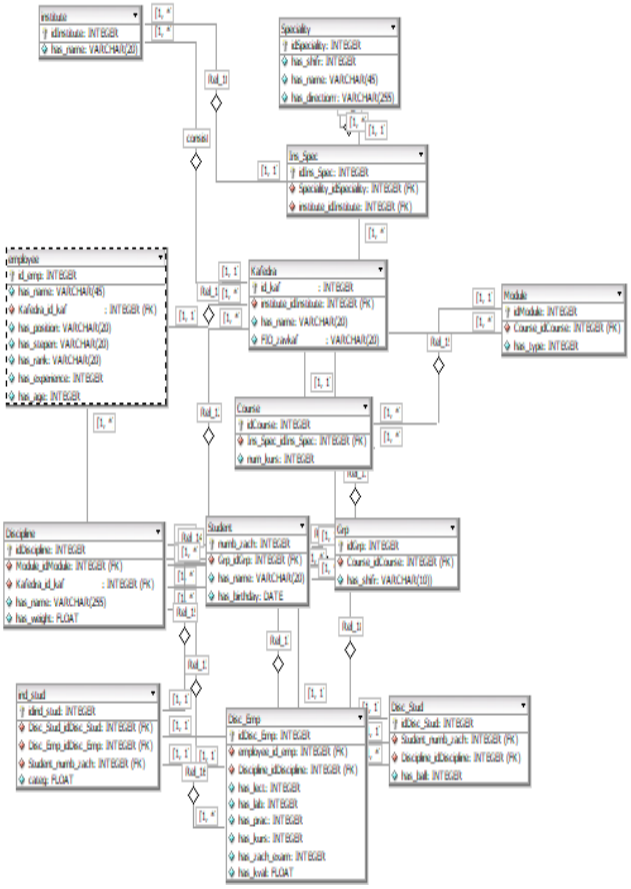

Fig. 3 Logical data model

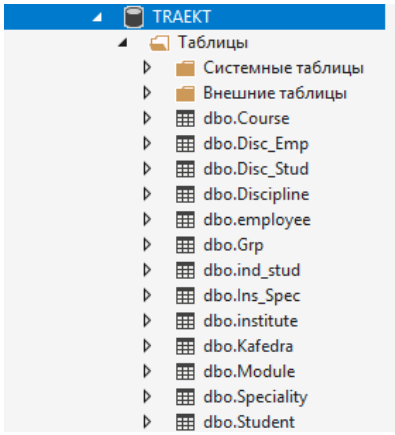

Fig. 4 List of database tables on the server

VI. STRUCTURE OF THE CLIENT APPLICATION FOR GENERATING DATA

There is no automation implemented in our National research Irkutsk state technical university related to monitoring the individual learning paths, so, in order to devise the structure details of the deep learning network and assess the quality of classification of individual learning paths, it is necessary to generate paths similar to the real ones.

The generation program is written in Python; data are generated using the "Mersenne vortex" algorithm, which provides high-speed generation and high quality of samples.

The generated data are saved as CSV tables, which rows are substituted into DDL and DML command templates.

To visualize the generated results, an interface has been developed that displays them as individual learning paths (Fig. 5).

Student	Ivanov A.M.	IBM-20-1	student in the discipline
Name of the discipline	Weight of the discipline	Evaluation of the discipline	
Programming	0.7	0.9	5
Mathematics	0.5	0.8	4
Student	Petrov N.N.	IBM-20-2	Evaluation of the student in the discipline
Name of the discipline	Weight of the discipline	Evaluation of the discipline	
Programming	0.7	0.9	4
Mathematics	0.5	0.8	5
Machine graphics	0.3	0.9	5

Fig. 5 Table representation of the generated individual learning paths

Based on the Python script, one need to create (or correct) an ontology with the FLUENT editor [7], which is used to make various queries to the knowledge base and use it for classification with a deep learning network. Since database and ontology adjustments can occur multiple times, we decided to automate this conversion using the ANTLR software tool.

VII. CONVERTING A DATABASE TO AN ONTOLOGY

Authors used own technique for converting the MS SQL database into the ontology in the language of the FLUENT editor using ANTLR tool [8-10]. This tool performs lexical and syntactic analysis of the text at input, as well as generate program code in C#. In addition to the Lexer and Parser classes, the Emitter class has been created, which allows us to generate code in the FLUENT language based on a script in the MS SQL language. The MS SQL grammar in ANTLR includes semantic attributes and semantic actions that used to generate code in C#.

ANTLR (from ANother Tool for Language Recognition) is a tool for creating compilers or interpreters of programming languages.

The syntax elements used by ANTLR lexical analyzer are listed below:

* "occurs 0 or more times";
+ "occurs 1 or more times";
? "maybe omitted";
( ) "grouping";
|selecting one alternative from several [10].

The description of an ANTLR grammar for a MS SQL language subset, first performed by the authors, is shown in Fig. 6. This grammar describes the rules for both terminal and nonterminal elements.

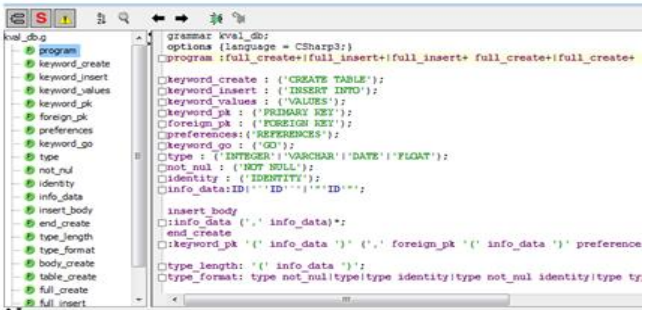

Fig. 6 Fragment of the grammar of a subset of the MS SQL language in the ANTLR system

VIII. DESCRIPTION OF THE ONTOLOGY ISOMORPHIC TO THE DATABASE OF INDIVIDUAL PATHS IN THE LANGUAGE OF THE FLUENT EDITOR

An ontology is a complete structural specification of a certain domain, its formalized representation that includes a dictionary of domain terms and a set of logical relationships (such as "element-class", "part-whole") that describe how these terms relate to each other [11].

The essential elements of any ontology are

- classes that describe concepts of the subject area;
- attributes that describe the properties of classes and their instances;
- instances (objects) containing specific attribute values;
- relations that are a subset of attributes, defining dependencies between classes.

FLUENT editor is a multi-functional and intuitive application that allows one to edit ontologies, visualize relationships between concepts and their instances, and execute queries to the knowledge base formed by all the elements of the ontology. A knowledge representation language is used in the system, which allows one to write explicit formalized descriptions of various subject areas. The natural language description is the main difference between FLUENT editor and other ontological editors and allows a much wider group of users to master the development of ontologies.

Statements of the form
"*Every institute is a thin.*"
define classes that are at the top level of the hierarchy (class names begin with a small Latin letter). Class attributes are defined by sentences of the form
"*Every institute has-name nothing-but (some string value)*.".
The relationships between classes are determined by sentences of the form
"*Every employee works-on a chair.*".
Instances of classes are defined by sentences of the form *"Insta is an institute and has-name equal-to 'CYBERNETICS'*",
instance names begin with a capital Latin letter.

The names of the database tables, their corresponding classes, attributes, instances, and relationships are shown in Table 2.

TABLE II. NAMES OF TABLES, CLASSES, OBJECTS, ATTRIBUTES, AND RELATIONSHIPS

Table	Class	Objects	Attributes	Relationships
institute	institute	Insta,Instb, …	has-name	is thing
speciailty	speciality	Speca, Specb,…	has-shifr, has-name, has-direction	belong-to
ins_spec	ins-spec	Isa, Isb,…	-	belong-to, consist
employee	employee	Empla, Emplb,…	has-name, has-position, has-stepen, has-rank, has-experience, has-age	works-on
kafedra	kafedra	Kafa, Kafb,…	has-name, has-name-zav	belongs-to
course	course	Coua,Coub, …	num-course	is-part-of
module	module	Moda, Modb,…	has-type	is-a-block
discipline	discipline	Disca,Discb, …	has-name	is-an-element, taught
grp	grp	Grpa,Grpb, …	has-shifr	consists-of
student	student	Studa, Studb,…	has-name, numb-recbook, has-birthday	is-a-member
disc_emp	disc-emp	Dea, Deb,…	has-lect, has-lab, has-prac, has-lkurs, has-zach-exam, has-degree	contains, assigned-to
disc_stud	disc-stud	Dsa, Dsb,…	has-ball	belong-to, taught-by
ind_stud	ind_stud	Inda,Indb,…	has-categ	for-stud, taught-by, learn

A fragment of the ontology (FLUENT document) generated using ANTLR is shown in Fig. 7. With this document, the FLUENT editor creates a hierarchy of classes, their instances, attributes, and relationships, as shown in Fig. 8. The constructed graph of the interconnections between classes, their instances, and relationships are shown in Fig. 9. The generated relational database tables and their isomorphic ontology will be used in the future to create and train a recurrent deep learning network. Various queries can be made to the ontology; they are typed in the Reasoner window in the Controlled Natural Language (CNL) [7]. One can see the query result on the screen. Example query: Which teachers work at the Department of computer engineering? (Fig. 10). Such queries in a language similar to English are easily constructed by users who are not programmers. In addition, queries in the FLUENT ontological editor allow one to derive new knowledge from existing facts, which is not possible in SQL.

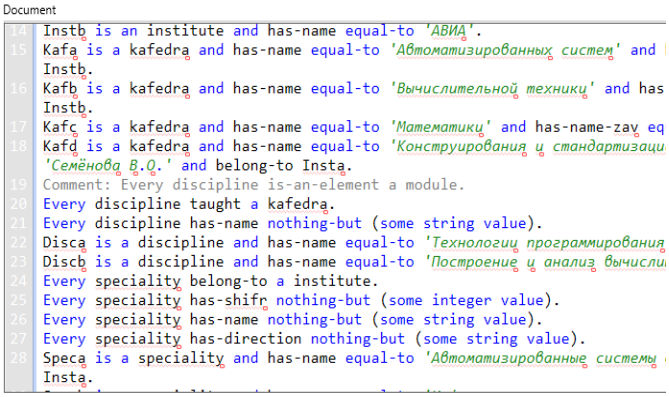

Fig. 7 An ontology fragment in the FLUENT editor

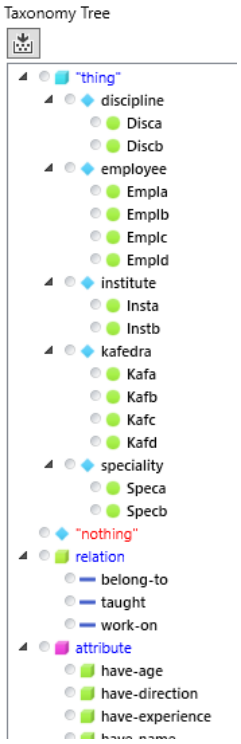

Fig.8 A hierarchy tree in the FLUENT editor

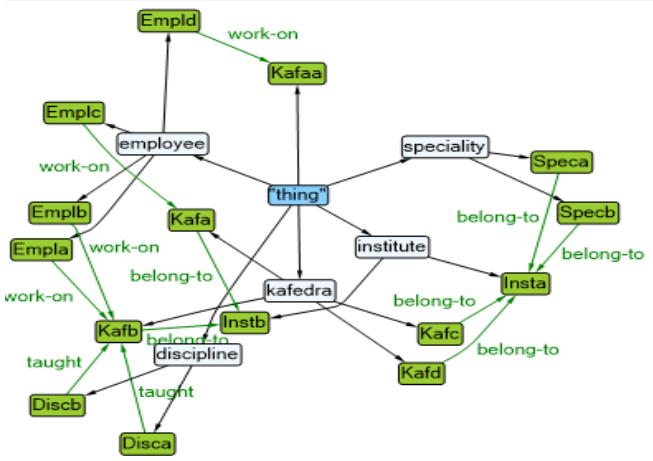

Fig. 9 Graph of interconnections between classes, relations and instances in the FLUENT editor

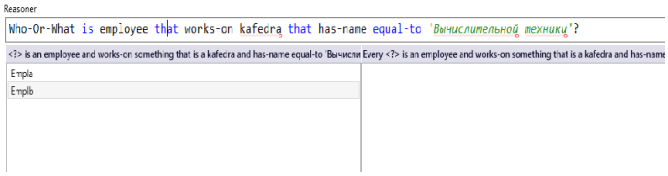

Fig. 10 An ontology query and its result

IX. Conclusion

The article deals with classification of individual learning paths, fundamentals of the qualimetric approach to evaluating learning paths, the structure of the database used and developed system are presented. A technique for converting the MS SQL database into the ontology in the language of the FLUENT editor using tool ANTLR has been described. The generated relational database and their isomorphic ontology will be used in the future to create and train a recurrent deep learning network.

The proposed method of generating data of individual learning paths of training will allow us to choose the best learning algorithms for the classification of learning paths using recurrent neural networks. This will allow us to evaluate the quality and effectiveness of this type of training.

References

[1] O.Yu. Zaslavskaya, O.J. Kravets, Features of Construction of the Individual Traectory Education to Computer Science on the Basis Dynamic Integrated Estimation of Level of Knowledge.UDN Bulletin, series Informatization of education, 2010, № 4.-pp.180-185

[2] S. Nicolenko, A.Kadurin,E. Arhangelskaya. Deep Learning. - Iimmersion in the World of Neural Networks. SPB Piter, 20018. – 470 p.

[3] Gruber, T. Towards Principles for the Design of Ontologies Used for Knowledge Sharing / T. Gruber // International Workshop on Formal Ontology. - Padova, Italy, 1993. – pp. 907 - 928.

[4] Gorshkov, S. Introduction to ontology modeling / S. Gorshkov // [Electronic resource] – URL: https://trinidata.ru/files/SemanticIntro.pdf (last accessed 13.05.2019).

[5] Azgaldov, G. G. Qualimetric expertise. Guide on organization of examination and implementation of qualitative evaluation. Book one. Organization of an expertise / G. G. Azgaldov, V. M. Marugin // SPb., Moscow: Russian Register, 2002. – 517 p.

[6] Azgaldov, G. G. Qualimetry: an introduction. Casebook with example for ANO "Agency for strategic initiatives to promote new projects": tutorial / G. G. Azgaldov, A. V. Kostin, V. V. Sadovov // M.: High school, 2011. — 143 p.

[7] Borgest, N. M. Ontology editor Fluent Editor / N. M. Borgest, A. A. Orlova // teaching manual for laboratory works. – Samara: publishing house of Samara University, 2017. - 44 p.

[8] ANTLR Official documentation. [Electronic resource]. – URL: https://pragprog.com/book/tpantlr2/the-definitive-antlr-4-reference (last accessed 15.05.2019).

[9] Tomassetti, G. The ANTLR Mega Tutorial / G. Tomassetti // in ANTLR, Language Engineering, Parsing [Electronic resource] – URL: https://tomassetti.me/antlr-mega-tutorial/ (last accessed 19.05.2019).

[10] Terence, P. The Definitive ANTLR Reference: Building Domain-Specific Languages / P. Terence // The Pragmatic Bookshelf, 2007. – pp. 376

[11] Gruber, T. Towards Principles for the Design of Ontologies Used for Knowledge Sharing / International Workshop on Formal Ontology. - Padova, Italy, 1993. – P.907-928