Specialized KD-Agent for Knowledge Ecosystems

Viktor Krasnoproshin
Faculty of Applied Mathematics and Computer Science
Belarussian State University
Minsk, Belarus
krasnoproshin@bsu.by

Vadim Rodchenko
Faculty of Mathematics and Informatics
Yanka Kupala State University of Grodno
Grodno, Belarus
rovar@grsu.by

Anna Karkanitsa
Faculty of Mathematics and Informatics
Yanka Kupala State University of Grodno
Grodno, Belarus
a.karkanica@grsu.by

***Abstract*—One of the base elements of any knowledge ecosystem is a software agent. The agent receives data about the internal events of the ecosystem, interprets data and executes commands that affect the environment. The paper proposes an option for the implementation of the specialized Knowledge Discovery agent (KD-agent). The input for the agent is the a priori dictionary of features and the training set. As the outcome of the agent activity previously unknown patterns are revealed and can be interpreted within the subject domain. The effectiveness of the proposed approach is demonstrated on the example of model data analysis.**

***Keywords*—knowledge ecosystem, data mining, supervised learning, training set**

I. Introduction

The knowledge ecosystem is a complex adaptive system including a database, a knowledge base and experts [1]. The development and implementation of such systems is one of the priority courses of information technologies growth and usage [2], [3].

The knowledge ecosystem is intended to provide high-quality interaction between objects for the effective implementation of the decision-making process. Typically, it includes technological core, critical interdependencies, knowledge agents and performative actions [1].

Knowledge agents receive and interpret data about internal ecosystem events and execute commands that have impact on the environment. The most important agents' properties are autonomy, social ability, reactivity and pro-activity [4].

The paper describes the process of constructing a specialized intelligent Knowledge Discovery agent (KD-agent). The agent's input data are the a priori dictionary of features and the training set. In automatic mode, the agent performs data analysis on which a set of informative ensembles of features are formed ensuring the separation of classes. The results of the practical usage of KD-agent on the example of model data analysis are described.

II. Knowledge Discovery in Databases

The development of novel and application of existing data mining methods and technologies is a promising avenue of knowledge ecosystems development and use.

The ongoing progress in the development of artificial intelligence technologies is largely due to the wide implementation of machine learning methods based on identifying empirical patterns in datasets [5].

During the learning process an intellectual system is provided with a set of positive and negative examples related by a previously unknown pattern. As a result of learning, a decision rule (algorithm) used to split the presented examples into positive and negative is generated [6].

Thus, machine learning methods traditionally construct practically useful algorithms (decision rules) that implicitly express empirical patterns. For example, the result of the Supervised Learning is a classification algorithm that is a certain practically useful "black box". This result, unfortunately, defies any interpretation within the subject domain.

Knowledge Discovery in Databases (KDD) is a process of discovering in the initial datasets a previously unknown, useful and interpretable patterns, which are further necessary for effective decision-making [7]. However, as it's shown above, machine learning methods do not fully satisfy all the KDD requirements. They do not allow to interpret the discovered patterns.

Formally, the KDD process includes five major stages and can be represented as follows (Fig. 1):

$$DW \xrightarrow{S1} TD \xrightarrow{S2} TS \xrightarrow{S3} DM \xrightarrow{S4} Ps \xrightarrow{S5} K$$

where DW and TD — data warehouse and a target dataset respectively; TS — training set; DM — Data Mining procedure; Ps — resulting set of patterns; K - – knowledge; S1 (Stage 1) — the stage of formulating the goal and objectives of the KDD process and the formation of a target dataset on which the search for patterns will be carried out; S2 (Stage 2) — the stage of data preprocessing and formation of a training set; S3

(Stage 3) — execution of the Data Mining procedure; S4 (Stage 4) -– building class patterns; S5 (Stage 5) -– patterns interpretation in terms of the subject domain.

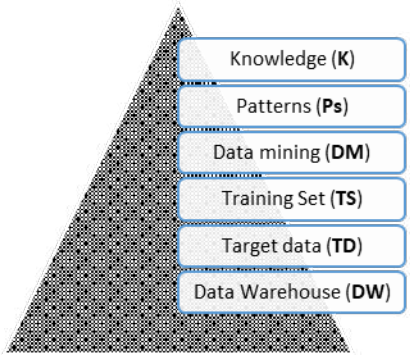

Figure 1. Patterns detection circuit.

Thus, the KDD process begins with the selection of an alphabet of classes, a set of observed features and the construction of an a priori dictionary of features (PDF). Each observed object is then represented as a vector of features from the PDF and, as an outcome, the training set is formed. Further, using the training set, estimates of informative value are calculated for all possible combinations (ensembles) of features from the PDF (in terms of the correct division of the pre-set classes).

After that, domains of classes (class patterns) for each combination of features are constructed. And on the results of the analysis of the mutual placement of patterns, the informativeness of the corresponding ensembles of features are estimated.

Therefore, as a result of the KDD process implementation, we are acquiring knowledge in the form of the informative significance of ensembles of features from the PDF. The knowledge thus acquired can be interpreted in terms of the subject domain, since each feature in any combination carries a specific semantic load.

III. Knowledge Discovery vs Supervised Learning

As noted previously, a classification algorithm (decision rule) is constructed on the basis of the results of the Supervised Learning procedure performance.

Traditionally, the learning process is reduced to the construction of decision rules that deliver the extremum of some functional. Therefore, decision rules families, generally, are selected a priori with accuracy up to parameters. In the learning process, specific values of the parameters which provide the extremum of the pre-set functional are determined.

It is considered that the dictionary of features is used not only for constructing a training set. It also defines a feature space in which the decision surfaces between classes are built.

On the basis of Machine Learning methods, it is possible to solve many applied problems that quite recently were considered non-trivial. In particular, impressive results have been obtained using the technology of artificial neural networks.

At present, neural network technologies allow to provide not only a high-level quality of learning, but also include for a nearly autonomous execution of the Supervised Learning procedure. However, application of artificial neural networks, as well as other Machine Learning methods, is limited to developing a classification algorithm. It turns out that a useful result of the entire resource-intensive process of training set preparing (about 80% of all costs) and processing is precisely the classifier. The classifier in fact is a “black box” that is not possible to further interpretation.

Thus, the main objective of machine learning methods is to build classification algorithms. In fact, this is a weak side of the approach. Although as an outcome it is succeed to learn how to separate class patterns, but at the same time there is no information of any kind about the properties of classes themselves.

An alternative to the data analysis of the training set can be an approach based on the idea of extracting some subsets from the a priori dictionary of features that would provide the separation of classes in a given feature subspace. Actually, features and their various combinations have varying informativity extent characterizing the properties of classes. Suppose the a priori dictionary contains n features. Obviously, 2n-1 of all possible combinations (ensembles) of features can be constructed [8]. If in such a set of combinations there is an ensemble by which the classes are well separating in a given feature subspace, then it can be stated that:

1) previously unknown patterns of classes properties are discovered;
2) these patterns can be interpreted in the subject domain terms;
3) based on the revealed properties, the problem of constructing a classifier becomes trivial.

To detect the described ensembles, it is proposed, initially, to build the domains of classes based on the data of the training set. Thereafter, in the corresponding feature subspace, we can calculate the estimates of their mutual placement.

Essentially, the process described above implements a typical procedure of knowledge discovery in dataset. On its basis, it is proposed to build an intelligent KD-agent that gets as an input a priori dictionary of features and a training set. Such KD-agent will automatically process the data and form the discovered patterns.

IV. Functioning of KD-agent

Within the classical for Machine Learning approach, the following statement of classification problem is adopted:

Let the objects descriptions X and the acceptable answers Y for objects classification are given. Suppose there is an unknown target dependency $y* : X \to Y$, *which values* $X^m = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ *are known only for the training set objects.*

It is necessary to construct an algorithm $a : X \to Y$ *that would approximate this target dependency not only on the objects of the finite set, but also on the entire set X* [9].

The solution to this problem is typically carried out in two stages. First, a certain family of algorithms is specified up to parameters. Then, in the learning process, the values of the parameters are determined that provide the extremum of the preselected functional.

The selection of algorithms model (family) $A = \{a : X \to Y\}$ is a non-trivial problem. Such a choice requires the participation of a qualified specialist. It means that learning is only carried out in an automated, but not automatic mode. Another serious disadvantage is that the resulting algorithm $a : X \to Y$ is a "black box" whose outcomes cannot be interpreted.

The application of the learning approach described above (alternative) avoids the mentioned disadvantages. The following modification of the problem statement is proposed:

Let the objects descriptions X and the acceptable answers of objects classification Y are given. There is an unknown target dependency $y* : X \to Y$, *which values* $X^m = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ *are only known for the training set objects.*

It is required to find feature subspaces where class patterns do not intersect.

Let the training set $X^m = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ be formed on the basis of the dictionary of features $F = \{f_1, \ldots, f_n\}$. Let $V = \{v_1, \ldots, v_q\}$ denote the set of all possible combinations (ensembles) of features from F. Obviously, V contains $q = \sum_{i=1}^{n} C_n^i = 2^n - 1$ subsets.

The algorithm of constructing feature subspaces $V^* = \{v_1^*, \ldots, v_k^*\}$, where class patterns do not intersect is as follows:

Step 1. In the set V, n combinations $V^+ = \{v_1^+, \ldots, v_n^+\}$, that contain one feature are being selected. For each individual feature, class patterns are built and their mutual placement is estimated. If the patterns do not intersect, then the feature is included in the resulting set V^*. The combinations that contain this feature are excluded from the set V. If the patterns intersect, then the feature is excluded from V.

Step 2. Let $V^\Delta = \{v_1^\Delta, \ldots, v_p^\Delta\}$ denote by the subset obtained as a result of the set V transformation at the previous step. For each individual combination from V^Δ, class patterns are built and their mutual placement is estimated.

If the patterns do not intersect, then the combination of features is included in the resulting set V^*. And all elements that contain this combination are being excluded from V.

If the patterns intersect, then the combination is excluded from V^Δ. The process is repeated until V^Δ becomes empty.

As a result of the analysis of all elements from $V = \{v_1, \ldots, v_q\}$ (possible combinations of features from the dictionary $F = \{f_1, \ldots, f_n\}$) a set $V^* = \{v_1^*, \ldots, v_t^*\}$ will be constructed, where $0 \le t \le q$.

On the basis of each separate ensemble-combination $v_i^* \in V^*$, we formulate a previously unknown, empirically revealed pattern: **in the feature space of the subset** v_i^* **the classes do not intersect**. It should be noted that within a specific applied problem, each combination of features v_i^* can be interpreted by a subject domain expert.

So, as an input, the KD-agent receives an a priory dictionary of features $F = \{f_1, \ldots, f_n\}$ and the training set $X^m = \{(x_1, y_1), \ldots, (x_m, y_m)\}$. Based on the above algorithm, agent forms the set $V^* = \{v_1^*, \ldots, v_t^*\}$, where $0 \le t \le q$ (Fig. 2).

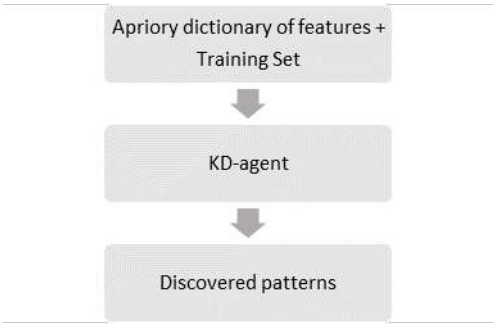

Figure 2. KD-agent workflow.

Let's note that the so constructed KD-agent satisfies all major characteristics subjecting to agents in multi-agent systems (autonomy, local representations, decentralization).

V. Application of the KD-agent

Let's demonstrate the efficiency of the KD-agent by the example of analyzing the training set data aiming to reveal hidden patterns.

Example. Let the given:

- number classes – **even** and **odd**;
- a priori dictionary of features F = {units, tens, hundreds, thousands, tens of thousands, hundreds of thousands, millions};
- training set of seven-bit integers, which contains 2000 even and 2000 odd numbers.

Table 1 shows the results of researching the intersection of class patterns based on the feature **units**, where

$$NE_i = \textit{Number of even}_i$$
$$NO_i = \textit{Number of odd}_i$$

$$a_i = \begin{cases} NM5_i + NNM5_i, NM5_i = 0 \vee NNM5_i = 0 \\ 0, NM5_i > 0 \wedge NNM5_i > 0 \end{cases}$$

$$Intersection = \frac{20000 - \sum_{i=0}^{9} a_i}{20000} * 100\%$$

Table I shows that even numbers are lack of 1, 3, 5, 7, 9 in the unit's digit, and odd numbers are lack of 0, 2, 4, 6, 8. In addition, the **units** feature provides an absolute separation of the classes **even** and **odd** since the $Intersection = 0\%$.

Table I
Experiment results for the feature **units**

Digit	Number of even	Number of odd
0	405	0
1	0	415
2	408	0
3	0	398
4	373	0
5	0	383
6	423	0
7	0	404
8	391	0
9	0	400

Table II
EXPERIMENT RESULTS FOR THE FEATURE **TENS**

Digit	Number of even	Number of odd
0	204	192
1	201	204
2	205	190
3	190	216
4	203	191
5	183	216
6	200	192
7	216	194
8	197	195
9	201	210

Table II shows the analysis results for the feature **tens**. It could be seen therefore that this feature has no the property of class separation since $Intersection = 100.0\%$.

Table III presents the analysis results for the feature **millions**. The table shows that this feature does not have the property of class separation neither, since $Intersection = 100.0\%$.

Table III
EXPERIMENT RESULTS FOR THE FEATURE **MILLIONS**

Digit	Number of even	Number of odd
0	201	211
1	212	211
2	193	187
3	174	190
4	189	191
5	210	181
6	208	204
7	196	210
8	218	212
9	199	203

Table IV shows the results of the analysis for all features from the a priori dictionary.

Table IV
EXPERIMENT RESULTS FOR ALL FEATURES

Feature name	Intercestion (%)
units	0.0
tens	100.0
hundreds	100.0
thousands	100.0
tens of thousands	100.0
hundreds of thousands	100.0
millions	100.0

Let's note that the algorithm running time spent on solving this problem was only 0.09 seconds.

VI. CONCLUSION

The paper presents the implementation variant of the specialized knowledge discovery agent (KD-agent). The input for such an agent is the a priori dictionary of features and the training set. As the outcome of the KD-agent activity previously unknown patterns are revealed and can be interpreted by experts of the corresponding subject domain. It is easy to see that the outcomes of the KD-agent's work can be further used by other agents of the ecosystem.

The effectiveness of the proposed approach is demonstrated on the example of the model data analysis.

REFERENCES

[1] Knowledge ecosystem [Electronic resource]. Available at: https://en.wikipedia.org/wiki/Knowledge_ecosystem (accessed 2021, March).

[2] C.W. Choo, N. Bontis, The Strategic Management of Intellectual Capital and Organizational Knowledge, Oxford University Press, 2002, 748 p.

[3] S. Russel and P. Norving, Iskusstvennyj intellect: sovremennyj podkhod [Artificial intelligence: a modern approach], Moscow: Williams Publishing House, 2006, 1408 p.

[4] AIportal [Electronic resource]. Available at: http://www.aiportal.ru/articles/multiagnt-systems/weak-and-strong-intelligent-agent.html (accessed 2021, March).

[5] P. Flach, Mashinnoe obuchenie. Nauka i iskusstvo postroeniya algoritmov, kotorye izvlekayut znaniya iz dannykh [Machine learning. Science and art of constructing algorithms that extract knowledge from data], Moscow: DMK Press, 2015, 400 p.

[6] MachineLearning [Electronic resource]. Available at: https://en.wikipedia.org/wiki/Machine_learning (accessed 2021, March).

[7] Data mining [Electronic resource]. Available at: https://en.wikipedia.org/wiki/Data_mining (accessed 2021, Feb).

[8] V.G. Rodchenko, Pattern Recognition: Supervised Learning on the Bases of Cluster Structures, XIII International Conference "Pattern Recognition and Information Processing", Minsk, BSU, Communications in Computer and Information Science, Springer, 2017, pp.106-113.

[9] Supervised learning [Electronic resource]. Available at: https://en.wikipedia.org/wiki/Supervised_learning (accessed 2021, March).

Специализированный KD-агент для экосистем знаний

Краснопрошин В.В., Родченко В.Г., Карканица А.В.

Одним из базовых элементов любой экосистемы знаний является программный агент. Находясь в среде экосистемы, агент получает данные о внутренних событиях, интерпретирует их и выполняет команды, которые воздействуют затем на среду. В статье предлагается вариант реализации специализированного knowledge discovery агента (KD-агента). Входными данными для агента являются априорный словарь признаков и обучающая выборка. В результате работы агента выявляются ранее неизвестные закономерности, которые могут быть проинтерпретированы экспертами-специалистами соответствующей предметной области. Эффективность предложенного подхода демонстрируется на примере анализа модельных данных.

Received 23.05.2021