DOI: https://doi.org/10.33216/1998-7927-2019-253-5-53-62

UDC: 004.942

DATA MINING TECHNIQUES FOR IOT ANALYTICS

Krytska Y.O., Biloborodova T.O., Skarga-Bandurova I.S.

ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ ДЛЯ IOT АНАЛІТИКИ

Критська Я.О., Білобородова Т.О., Скарга-Бандурова І.С.

Data mining (DM) is one of the most valuable technologies enable to identify unknown patterns and make Internet of Things (IoT) smarter. The current survey focuses on IoT data and knowledge discovery processes for IoT. In this paper, we present a systematic review of various DM models and discuss the DM techniques applicable to different IoT data. Some data specific features were analyzed, and algorithms for knowledge discovery in IoT data were considered. Challenges and opportunities for mining multimodal, heterogeneous, noisy, incomplete, unbalanced and biased data as well as massive datasets in IoT are also discussed.

Keywords*: Data Mining, Internet of Things, IoT, Knowledge Discovery in Database, KDD, massive data set*

Introduction. IoT applications generate more than 2.5 quintillion data bytes daily [1]. To convert this data into knowledge, data mining systems are increasingly in demand. Data mining (DM) enables to find and discover novel, interesting, and useful patterns from large data sets and generate new knowledge from information obtained from IoT devices. However, basic data mining algorithms and technologies are not quite sufficient for IoT framework. So, it becomes a great challenge to collect, analyze and manage IoT data as well as to generate and update data mining algorithms for IoT purposes. In this paper, we discuss some DM approaches applicable for IoT data. An important aspect of DM of the IoT-based system is the effective structure of the system, which should take into account security, data privacy, data sharing mechanisms, scalability, etc. Such a DM system for IoT includes data acquisition devices, raw data properties, extraction levels, processing, data analysis, it is necessary to take into account the properties of the IoT devices when planning DM for IoT [5]. Technically, every IoT thing can create data, but technical issues and challenges on how to handle this data and how to obtain useful information have still emerged.

The general purpose of any DM process is to build a best predictive or descriptive model of a large amount of data that not only fits or explains it but is also able to generalize to new data [3, 4]. It is assumed that the concept of DM for IoT will stimulate business models for IoT. Based on a broad understanding of DM functionality, data mining is the process of finding interesting knowledge from large amounts of data stored in any database, data repositories, or other data repositories.

Data mining techniques for IoT based applications has been widely presented in literature for different decision tasks, such as supervised and unsupervised learning for IoT applications [5, 14, 32], frequent pattern recognition and association analysis [4, 19, 21], massive IoT data mining [22, 24], stream data mining [36], etc.

The detailed surveys on the approaches, tools and techniques employed in existing for IoT data mining can be found in [2, 23]. Practical approaches in massive data processing for IoT applications are present in [31, 32, 33] for parallel and distributed data processing.

This survey focuses on IoT data and knowledge discovery processes for IoT. Our main contribution in this paper is that we targeted on data specific features and selected some well-known algorithms best suited for knowledge discovery in different IoT applications.

1. IoT data characters

IoT devices and sensors have two general limitations that must be considered when designing and planning the operation of IoT data mining systems:

- Limited energy resource device.
- Limited device memory.

The instability of the network connection and availability of the thing due to the unpredictable mobility of devices, different battery discharge rates, equipment failures and lack of a priori knowledge of the hardware and software characteristics of devices [6-8].

The power source must match the data, i.e. be sufficient for IoT computations. Storing information leads to the expenditure of battery energy. The solution to this problem is provided by storing and performing computational operations using remote IoT computing resources, such as a server and / or cloud. So far as IoT systems create a huge amount of dynamic data, analysis and extracting useful information from this data with DM can facilitate the automation of intelligent decision making.

IoT data can be:

- multimodal and heterogeneous;
- noisy and incomplete;
- unbalanced and biased;
- dependent on time and location;
- dynamic, different data quality;
- almost always require real-time analysis.

Given that IoT data is the basis for extracting knowledge, it is important to have high quality information. This condition can directly affect the accuracy of knowledge extraction.

Figure 1 shows an overall level for transformation of data and depicts a level of services where big DM for IoT is applicable.

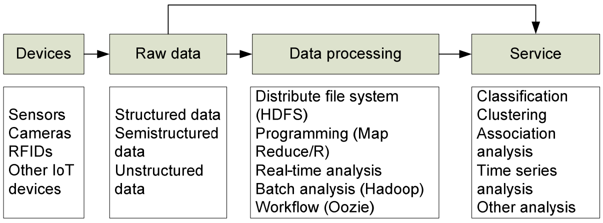

Fig. 1. Big data mining based on IoT (Adapted from [2])

2. Basic idea of using data mining for IoT

One of the most important questions that knowledge discovery in databases (KDD) and data mining technology can solve is how to transform the data generated or captured by IoT into knowledge that serve to the environment and people.

The main characteristics of the source data of IoT-based system are the following [2]:

1. They are really big data.

2. Heterogeneity of the sources being combined and the types of data: the data of the IoT-based system may include several data sources, for example, data from sensors, historical data, which may also have different formats: numerical, categorical, textual, binary, etc.

3. The complexity of recoverable knowledge: due to heterogeneity and a large amount of data when extracting knowledge, it is necessary to analyze their properties and the interrelation of various data sources. These characteristics require special attention in the process of DM of the IoT-based system for obtaining an effective and high-quality result.

In the process of extracting useful knowledge, there are the following issues.

1. Data extraction: data can be combined from various sources, they are diverse and heterogeneous, and noisy.

2. Uncertainty and incompleteness of data: compliance with data security and confidentiality causes uncertainty and incompleteness of data in the extraction of useful knowledge.

To solve these problems, approaches and methodologies are being developed that try to minimize their consequences. Tracking and detection of data errors, preprocessing filtering, and data reduction mechanisms are used. To combine data from several sources, parallel programming models are used, for which classical approaches to DM are adapted.

The selection of models depends on the area of IoT in which they are applied. For example, in ecology: pollution prediction, anomaly detection, prediction and interpolation of missing events are common. In medicine, traditional models used to predict a patient's conditions can include his history, clinical data as input along with real-time status monitoring data. It is also important to consider that IoT includes temporary and massive data.

The merging of data or data fusion is associated with combining data from different sources so that the information obtained has less uncertainty than would be possible when these sources were used individually. The term "reducing uncertainty" in this case may mean more accurate, more complete or more reliable, or refer to the result of an emerging presentation based on combined information.

The DM process for IoT is similar to the base one, but there are some big differences. The process of extracting useful patterns from raw data is known as Knowledge discovery in databases (KDD). It is illustrated in Fig. 2.

The KDD process takes raw data as input and provides statistically significant patterns found in the data (i.e., knowledge) as output. From the raw data, a subset is selected for processing and is denoted as target data. Target data is preprocessed to make it ready for analysis using DM algorithm. Data mining is then performed on the preprocessed (and transformed) data to extract interesting patterns. The patterns are evaluated to ensure their validity and soundness and interpreted to provide insights into the data.

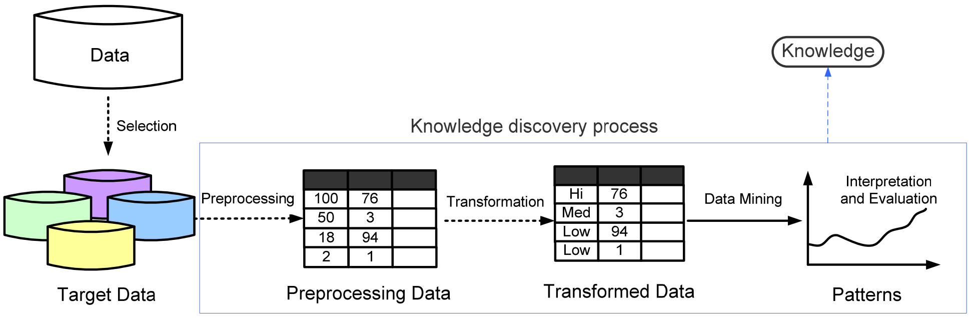

Fig. 2. Traditional KDD process

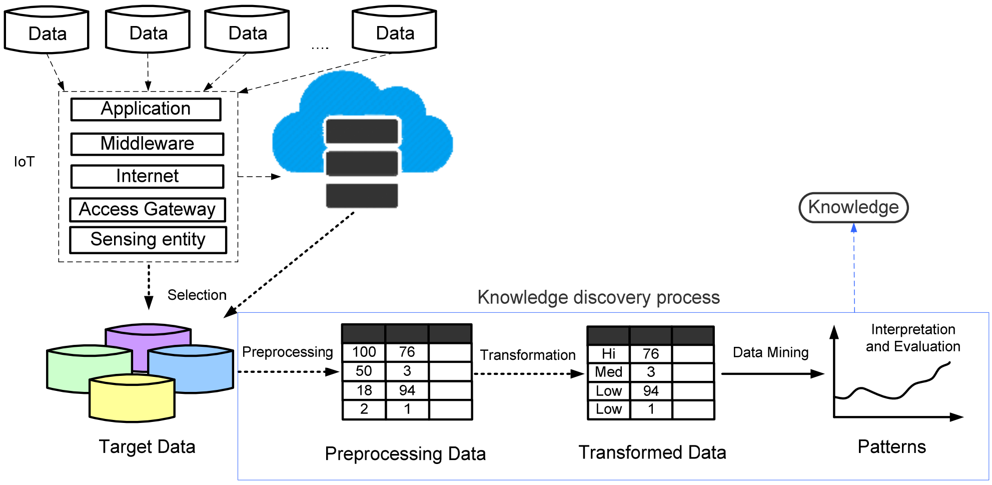

Fig. 3. KDD process for IoT data

Based on the DM and IoT overview, the data mining in IoT process is as follows (see Fig. 3): DM for IoT begins with the first step of capturing data generated from IoT devices which includes: Sensor networks, Actuators, Wireless Sensor Network (WSN), Wireless Sensor and Actuator Network (WSAN), Radio Frequency Identification (RFID) Tags, Cameras, GPS etc.

To store and analyze such large amount of data, data warehouses are used where data preprocessing (cleaning the data (removing noisy, inconsistent and incomplete data), vectorization the data), data transforming which includes converting the data into the forms appropriate for data analyzing, and data reducing are performed.

Next step is selecting an appropriate DM methodology for converting the preprocessed data into knowledge.

KDD, when applied to IoT, will convert the data collected by IoT into useful information that can then be converted into knowledge.

In many cases, only small scale data from IoT systems can be mined. Therefore, it is a big challenge to implement existing DM techniques to a large scale IoT distributed systems [9].

3. Applying data mining algorithms for IoT data

To determine which algorithm to use for a particular task, we need to first define the task and aim of analysis. Some of tasks include finding unusual data points, predicting values or categories, structures discovery, feature extraction and more.

Table shows some examples of using data mining algorithms for IoT data [10, 11].

3.1. Classification for IoT

Classification is an important technique in DM that assigns items in a collection of target categories or classes. Classifying the data sets into different categories enables to understand the data more easy. There are two big categories of classification, they are supervised and unsupervised learning that are widely used in IoT data mining [12]. The goal of supervised learning is to predict the corresponding output vector for a given input

Table

Application of DM algorithms for IoT data

DM algorithm	Goal	Data Source
Classification	Device recognition	RFID
	Traffic event detection	GPS, smart phone, and vehicle sensor
	Parking lot management	Passive infrared sensor
	Inhabitant action prediction	RFID, sensor, video camera, microphone, wearable kinematic sensor, and so on
	Inhabitant action prediction	Video camera
	Inhabitant action prediction	Microphone
	Physiology signal analysis	Wireless ECG sensor
Clustering	Network performance enhancement	Wireless sensor
	Inhabitant action prediction	X10 lamp and home application
	Provisioning of the needed services	Raw location tracking data
	Housekeeping	Vacuum sensor
	Managing the plant zones	GPS and sensor for agriculture
	Relationship in a social network	RFID, smart phone, PDA, and so on
Frequent Pattern	RFID tag management	RFID
	Spatial colocation pattern analysis	GPS and sensor
	Purchase behavior analysis	RFID and sensor
	Inhabitant action prediction	RFID and sensor
Anomaly Detection	Smart Traffic	GPS, smart phone, and vehicle sensor
	Smart Environment	Wireless sensor, smart phone
	Traffic Prediction	GPS, smart phone, and vehicle sensor
	Finding Anomalies in Power Dataset	RFID, wireless sensor
Hybrid	Inhabitant action prediction	RFID and sensor

vector. Tasks in which the output label value is discrete are known as classification problems. Classification assumes some prior knowledge to guide the partitioning process to construct a set of classifiers to represent the possible distribution of patterns.

Generally, the classification task can be defined as follows: for given a set of labeled data *L* and a set of unlabeled data *NL*, we need to find a classifier or set of classifiers (i.e., the hyperline or prediction function) for *NL* using the set of labeled data *L*.

The use of classification methods is a solution to the problem of uncertainty and incompleteness of IoT data. In this context, the use of DM always includes solving two related tasks: defining regular links between data elements and using these patterns to solve classification problems: predicting the values of some elements from known values of other elements.

There are a vast number of classification methods, they also includes decision tree learning, naïve Bayes classifier, k-nearest neighbor classifier, classification with neural network and regression methods such as linear regression and logistic regression, etc.

3.2 Clustering for IoT

One of the main goals of unsupervised learning is the process of identifying similar cluster patterns in the input data, called clustering [12]. In addition, the goal of DM may be to open a useful internal representation for the input data by preprocessing the original input variable, to transfer it to a new space of variables [13].

One of the most important parameter that needs to be determined during the clustering is a measure of similarity (or dissimilarity) between individual objects that are clustered [15]. One of the criteria to measure the similarity between two vectors x_1 and x_2 in d-dimensional space is the Euclidean distance

$$d(x_1,x_2)=\|x_1-x_2\|=\sqrt{\sum_{r=1}^{d}(x_{1r}-x_{2r})^2}, \qquad (1)$$
$$d(p,q)=\sum_{k-1}^{n}(p_k-q_k).$$

Many clustering algorithms have been developed for data analysis, which can be grouped into the following main categories: Partitioning-based clustering; Hierarchical clustering; Grid-based clustering; and Density-based (DB) clustering algorithms. It should be mentioned that characteristics of data streams do not allow the use of traditional DB clustering. Recently, many DB clustering algorithms have been extended tailored to data streams. The main idea of these algorithms is to use the DB method in the clustering process and at the same time overcome the limitations that are determined by the nature of the data flow. There are two broad groups of DB clustering algorithms called density micro-clustering and density grid-based clustering algorithms.

Clustering helps to solve the following IoT data analysis tasks:

- Processing of data of high dimension. Often complex concepts of the real world are accompanied by a large number of functions. This strengthens the assessment tool (for example, a classifier) to deal with a large number of functions for learning and in order to be able to generalize afterwards. Inside these functions it is often either redundant or irrelevant, and their use usually affects the complexity and the need for computational resources.
- Cluster heterogeneity. Distance-based clustering algorithms tend to find spherical clusters with the same size and density. Clustering algorithms that can detect clusters of arbitrary shape, size, density, and data coverage help to gain a deeper understanding of the various correlations between functions, which, in turn, can greatly facilitate the decision-making process.
- Interpretable results. The high dimension of the data space is cumbersome for rendering methods.

Clustering is also widely used to handle streaming data [14].

3.3. Frequent Pattern Mining for IoT

Recently, much attention has been paid to new promising methods for extracting interesting knowledge from data from the IoT-based system. DM algorithms that have low computational complexity are being developed [16-18]. The processes of forming frequently occurring patterns, creating association rules are computationally simple in this respect and are often used as methods for finding interesting knowledge.

The disadvantage of this analysis is the detection of patterns, rules that do not contain meaningful information.

Since IoT-based systems generate large amounts of data, it is necessary to use appropriate measures of significance, which have a strong correlation between the data, to search for frequently occurring patterns, associations.

The basic prerequisites of the model for the effective detection of commonly occurring patterns in IoT data are [19]:

1. Determination of the relevant significance parameters for the detection of patterns that meet the downward closure property; when all subsets of the frequent set of features are frequent, to reduce the search space.

2. Compactness of the structure of the model, obtained by using the distributed and parallel methods of DM.

3. Adaptability of the model structure for effective analysis of the latest relevant information and extraction of relevant patterns in the data. To fulfill this condition, the optimal size of the data window is determined, which helps to avoid the rapid obsolescence of information.

The dimension of the rule space depends on the minimum threshold of parameters defining the significance of the rules. If the minimum threshold is set high, then we can extract valuable knowledge. On the other hand, at a low minimum threshold of the rule significance parameter, an extremely large number of association rules are generated, most of which are non-informative. In this case, the actual correlation in the data is hidden among a huge number of insignificant rules [20].

Frequent pattern mining (FPM) in some domain often involve real challenges arise from their nature and the field of application. Frequent pattern and associations rules involve many items that hard to interpret and generate a lot of outcomes. In many cases, the obtained association rules can either be too obvious, or contradict a priori knowledge, or contain redundant information. The task of FPM and mining association rules is to generate minimal set of rules providing complete coverage of outcomes with objective parameters, such as support and confidence greater or equal than some pre-specified thresholds of minimum support and minimum confidence, respectively. FPM process includes several steps.

Data transformation into the nominal scale is among the first steps in FPM. The transformation process realizes different goals depending on the approach. In a normative-oriented approach, the reduction of indicators makes it possible to determine the value of a variable with respect to certain generally accepted norms, or to compare the results, giving a definition of the value of a variable with respect to the other values.

When the criterion-oriented approach is given, the value shows the percentage of compliance with the value of the variable to a specific criterion.

Next stage is sorting the rules according to the class and reducing the number of rules by the elevator parameter as follows.

Step 1: set formation L_1 of one-item sets c_1, that often meet and determine their support;

Step 2: set formation L_k k-item sets, that often meet. Each member of the set has a set of ordered $(i_j < i_v$, if $j<v)$ itemes F and he support value of the set $supp_F > supp_{min}$:

$$L_k = \{(F_1, supp_1), (F_2, supp_2), ..., (F_q, supp_q)\}, \qquad (2)$$

where $F_j = \{i_1, i_2, ..., i_k\}$.

The definition from the set L_k of k-item sets, corresponding to a certain minimum threshold value of support.

Step 3: based on the specific sets of element sets from step 2, the formation of the set C_k rules k-item sets is potentially often encountered. Each member of the set has a set of ordered $(i_j < i_v$, if $j<v)$ itemes F and a support value of the set of *supp*.

Formation of a set k-item sets into frequent sets. According to this, the integration into k-item rules of *(k-1)*-item sets, s carried out, often encountered. Each rule $R \in C_k$ is formed by adding v to an *(k-1)*-item set *v*, that frequently occurring item with another *(k-1)*-item set *q*, that frequently occurring.

Step 4: reduction of all uninteresting rules using measures of determining the interestingness of rules.

3.4. Association analysis

The results of DM are certain patterns and trends whereby we have to find out the interesting patterns best suit to our needs. However, after applying the some DM methodologies for IoT environments, a large number of patterns are evaluated. Many of these patterns are non informative and that is why not interesting for further analysis. Patterns become interesting when they are unknown till yet and not expected. With this purpose, associative analysis can be used.

The goal of associative data analysis is to identify associations between input and output data, identify the most specific factors for the qualitative separation of variables into classes, and quantitatively describe the relationship between these events. When defining associations in the data, a large number of rules are usually obtained. To determine their information value, it is necessary to use methods to reduce their number and determine from them potentially interesting ones.

In general case, association analysis algorithms generate a huge number of items and can produce up to hundreds of association rules. An association rule is an implication expression

$$R : X \rightarrow Y,$$

where X denoted antecedent and Y denotes consequent $X \cap Y = \varnothing$. Both X and Y are considered as a set of conjuncts of the form c_1, c_2 ..., c_k. The strength of the association rule is measured in terms of its support (s), confidence and interestingness.

For pair of rule-candidates, binary variables R_1 and R_2 the lift is equivalent to interest factor, which is defined as follows:

$$I(R_1, R_2) = \frac{s(R_1, R_2)}{s(R_1) \cdot s(R_2)}. \qquad (3)$$

The measure of interestingness in this case can be interpreted as follows:

$$I^1(R_1, R_2) \begin{cases} = 1, & \text{if } R_1 \text{ and } R_2 \text{ are independent,} \\ > 1, & \text{if } R_1 \text{ and } R_2 \text{ are positively correlated,} \\ < 1, & \text{if } R_1 \text{ and } R_2 \text{ are negatively correlated.} \end{cases} \qquad (4)$$

In order to increase the information importance of rules, it is necessary to reduce their number and focus on potentially interesting ones. Further exploration of interestingness leads us to discovering different subjective and probabilistic measures of interestingness. To determine the interestingness of the rule, various probabilistic measures are used: support, confidence, Goodman-Kraskal, Pyatetsky-Shapiro, Laplace, etc. In the present study, for reducing number of rules we applied three level technique proposed in [21] beginning with detection of deviations in data, then testing of differences among adjusted attributes and finally, quantifying the interestingness of association rules.

1. Detecting deviations in data is performed as follows.

The every conjunct cj from association rule set is represented in the form <A = V>, where A is an item name (attribute), Dom (A) is the domain of A, and I (value)$\in$Dom (A). Degree of deviation is defined as deviation between two conjuncts $\Delta(c_i, c_j)$ and is calculated on the basis of the comparison between the items of the two conjuncts. For conjuncts c_i, c_j deviation of c_i with respect to c_j is defined as a Boolean function as follows:

$$\Delta(c_i, c_j) = \begin{cases} 0, & \text{if } A_i = A_j \text{ and } V_i = V_j, \\ 1, & \text{if } A_i = A_j \text{ and } V_i \neq V_j. \end{cases} \tag{5}$$

2. The differences among adjusted attributes can be calculated using the following formula:

$$\bar{d}(R_1, R_2) = \begin{cases} 0, & \text{if } |R_1| = |R_2| \forall c_i \in R_1, \exists\, c_j \in R_2, \text{ that } \Delta(c_i, c_j) = 0, \\ 1 & \forall c_i \in R_1, \neg\exists\, c_j \in R_2, \text{ that } \Delta(c_i, c_j) = 1, \\ \dfrac{\sum\limits_{c_i \in R_1, c_j \in R_2} \min \Delta(c_i, c_j)}{|R_1|}, & \text{othervise,} \end{cases} \tag{6}$$

where R_1 and R_2 are considered as two sets of conjuncts c_i and c_j.

Parameter value $\bar{d}$ = 0 indicates that R_1 and R_2 are identical, $\bar{d}$ = 1 indicates the maximum deviation between rule sets, and the other $\bar{d}$ values between 0 and 1 are defined as a transient deviation.

3. Quantifying the interestingness of association rules

Let $R_1 : X_1 \rightarrow Y_1$ and $R_2 : X_2 \rightarrow Y_2$ be two association rules, then interestingness of a rule R_1 with respect to the rule R_2 is calculated as follows:

$$I^{II}(R_1, R_2) = \begin{cases} 0, & \text{if } \bar{d}(X_1, X_2) = 0 \text{ and } \bar{d}(Y_1, Y_2) = 0, \\ \left(\min\limits_{S \in R} \bar{d}(X_1, X_2) + \bar{d}(Y_1, Y_2)\right)/2, & \text{if } \bar{d}(X_1, X_2) \geq \bar{d}(Y_1, Y_2), \\ \left(\bar{d}(X_1, X_2) + \min\limits_{S \in R} \bar{d}(Y_1, Y_2)\right)/2, & \text{if } \bar{d}(X_1, X_2) < \bar{d}(Y_1, Y_2), \\ 1, & \text{if } \bar{d}(X_1, X_2) = 1 \text{ and } \bar{d}(Y_1, Y_2) = 1. \end{cases} \tag{7}$$

According to formula (7), I^{II} = 0 indicates that R_1 and R_2 are identical, I^{II} = 1 denotes maximum deviation between R_1 and R_2. Other cases indicate different deviations in the interestingness of association rules. To select interesting rules the user should specify the threshold of their interestingness. The anti-monotone property based on the threshold of the measure of interest can be applied to reduce the dimension of the resulting rule set. The anti-monotone property is that the measure of the interest of any set of elements should not exceed the minimal measure of interest of any of its subsets. This property greatly facilitates the mining rules.

4. Mining of Massive Datasets

IoT systems include multiple heterogeneous networked embedded devices that generate massive amounts of data. Massive Data IoT leads to different issues in processing and DM [22]. Figure 4 presents the main challenges associated with processing and mining massive data sets. The large amount of data, the high transfer rate and the variety of properties of large IoT data necessitate a new requirement for intelligent analysis of such data and the diversity in data sources is also a problem [23].

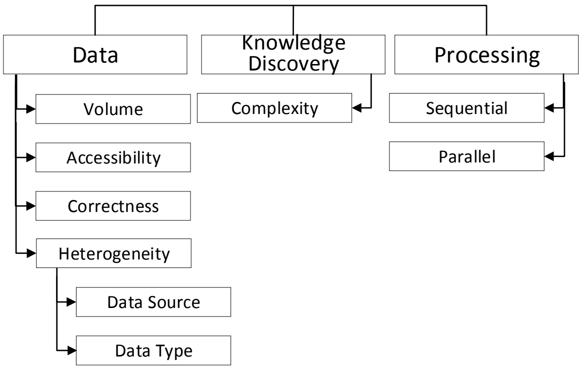

Fig. 4. Data mining issues in IoT

In addition, compared to small data sets, massive data sets contain more anomalies and ambiguities that require additional preprocessing steps [24]. Another problem is to extract accurate and useful information from large volumes of diverse data.

In accordance to [25], the massive data are generally collected from different heterogeneous sources (e.g., video cameras, sensors, RFID, other IoT devices, people, etc.) providing heterogeneous sensing data (e.g., text, video, sound). In this context, heterogeneous data processing (e.g., fusion, classification) brings new challenges and open new possibilities for systems. Obviously, these random variables from heterogeneous sensors have different probability distributions.

Define z_n as the data from the n-th sensor and $Z := \{z_n\}_{n=1}^{N}$ as the heterogeneous data set, the margins $\{z_n\}_{n=1}^{N}$ are generally differently or heterogeneously distributed.

In many IoT applications, datasets are often modeled as multi-sensor data fusion, distribution estimation or distributed detection. For detection, this tasks joint probability density function $f(Z)$ of the heterogeneous data set Z is needed to get from the marginal probability density function $\{f(z_\partial)\}_{n=1}^{N}$.

In these cases, one often uses simple models such as the product model or multivariate Gaussian model, which lead to suboptimal solutions [26]. Other approaches are based on copula theory, to tackle heterogeneous data processing in IoT. In copula theory, it is the copulas function that couples' multivariate joint distributions to their marginal distribution functions, mainly thanks to the Sklar theorem.

Sklar' theorem can be present as follow. Let F be an N-dimensional cumulative distribution function with continuous marginal probability density function F_1, F_2,

..., F_N. Then there is a unique copulas function C such that for all $z_1, z_2, ..., z_N$ in $[-\infty, +\infty]$

$$F(z_1, z_2, ..., z_N) = C(F_1(z_1), F_2(z_2), ..., F_N(z_N)) \tag{8}$$

Next, the probability density function can be obtained by the N-order derivative of (8)

$$\begin{aligned} f(z_1, z_2, ..., z_N) &= \frac{\partial^N}{\partial_{z_1}, \partial_{z_2}, ..., \partial_{z_N}} C(F_1(z_1), F_2(z_2), ..., F_N(z_N)) \\ &= f_p(z_1, z_2, ..., z_N) c(F_1(z_1), F_2(z_2), ..., F_N(z_N)) \end{aligned} \tag{9}$$

where $f(z_1, z_2, ..., z_N)$ is the product of the marginal probability density function $\{f(z_\partial)\}_{n=1}^{N}$ and c(·) is the copula density weights the product distribution appropriately to incorporate dependence between the random variables. The technique on the selection of proper copula functions is presented in [27].

4.1. CRISP-DM methodology for IoT domain

CRISP-DM (Cross Industry Standard Process for Data Mining) is an interdisciplinary data mining standard [28]. CRISP-DM uses six steps for data mining.

1. Understanding the business - involves understanding how the goals and requirements of the project are related to the business goals, formulating the problem of DM based on this understanding.

2. Understanding data - includes the initial stages of collecting and analyzing data to obtain initial information about their properties, determining the quality of data, identifying preliminary patterns and forming hypotheses.

3. Data preparation - includes the definition and execution of all actions that convert the raw data into the final data set. The stage includes the selection of tables, observations, variables, as well as conversion and data cleansing that are compatible with the modeling methods used.

4. Modeling - selection, application, optimization of modeling methods.

5. Evaluation - the constructed models are tested and, using selected criteria, their effectiveness is evaluated.

6. Deployment - includes the organization and presentation of knowledge generated by the model in an easily interpretable form for the end user.

Consider the steps of CRISP-DM in the context of the IoT in accordance with Data Science for IoT - The Problem Solving Methodology. In the context of IoT, solving a problem means solving the original problem and providing incremental feedback.

The preparation stage, unlike the standard process, must take into account the diversity of data sources and the architecture of IoT systems.

Proceeding from this, at this stage the following processes are distinguished, which should be carried out iteratively:

- definition of data requirements;
- IoT architecture design;
- collection, cleaning, intelligence data analysis;
- continuous improvement.

The determination of the necessary data is carried out taking into account the scope of IoT usage. This could be IoT for health and healthcare, smart homes and cities, smart transportation, industrial, energy systems, etc.

When designing architecture, it is necessary to take into account the technology of IoT systems, which includes sensors, networks and analytical tools.

There are several factors that influence the choice of components when solving a specific problem. The choice is determined by the accuracy and reliability, availability and security, data transfer speed, energy efficiency.

The selected constituent elements will determine some characteristics of the data and, accordingly, analytical tools.

This is followed by the collection, purification and intelligence analysis of available data. This process is typical of CRISP-DM technology and allows you to isolate additional information based on the available data, for example, incorrect, inappropriate operation of one of the network devices, problems with receiving, transmitting data. This information is used to refine the IoT architecture until an optimal solution is reached.

The modeling stage is the stage of building a model, evaluating its effectiveness and the quality of solving the problem. The stage includes the following processes:

- model design to solve a specific problem;
- model evaluation;
- model and architecture deployment.

As in the preparation stage, these processes should be carried out iteratively, until the optimal parameters of the model are reached.

Evaluation of the model is carried out using classical statistical methods and parameters. Also, it should be evaluated in terms of solving the problem. If the model does not improve the basic state of the problem, then iteration of all stages is necessary, starting from the preparation stage. Since some assumptions about the data could be erroneous.

After obtaining an optimal assessment of the model, the architecture and model are deployed.

All the above steps in the context of IoT require continuous improvement. This is due to the rapidly evolving nature of IoT technology, Data Mining methods. Also, this is due to the problem being solved, which can also change and, therefore, the models used are changed to correct it. The continuous improvement phase includes the following iterative processes:

- feedback and understanding;
- specification of the IoT architecture;
- refinement model;

- deployment of a new model and architecture.

Also, depending on the context, there are several elements that can stimulate the improvement of IoT, for example, such as:

- change in performance of the current architecture and model;
- additional user needs;
- the emergence of new tools, methods, algorithms that can help solve the problem in a cheaper or faster way;
- the emergence of new data streams.

As it was mentioned above, IoT-based systems generate large amounts of data. Often, they are presented in the form of time series, analyzed in real time, which determines the methods of intellectual analysis at the modeling stage. The main task of time series analysis is the detection of anomalies. For this, the classical classification models used to detect anomalies are most often used. However, new approaches are being developed, which show good results for analyzing streaming data by fusion data from several sources.

4.2. Map reduce

In IoT applications, mass data processing such as MapReduce is constructed for parallel and distributed data processing [29]. Querying and reasoning for data can be adapted to large data is a more flexible approach.

One of the most popular parallel processing methods in cloud platform is MapReduce [30] and its open source implementation Hadoop for cloud-based parallel or distributed data processing. For the parallelization, scalability, load balancing, and fault-tolerance is MapReduce is widely used in cloud platforms for query processing for data analysis.

The MapReduce disadvantage does not directly support more complex operations such as fusion. More research on high-level, declarative management of complex data such as RDF is required for massively parallel processing of IoT data in the cloud.

1) Parallel processing methods for complex operations: a processing framework is used for massive data processing, incremental calculation, and iterative processing. The framework is implicitly used to synchronize the parallel programs execution without any user specification for events and trigger reactions to process the data. The Selective Embedded Just-InTime Specialization (SEJITS) [31] executes complex analytic queries on massive semantic graphs in big-data analytics.

2) Parallel processing methods for semi-structural data: for the RDF data processing task, effectiveness and tunable data partitioning framework SPA [32], that use at distributing processing of big RDF data, is presented to fast processing support of different size as well as complexity. A MapReduce framework is designed to carry out SPARQL query processing. Thus, RDFS reasoning can be involved in deductive databases and thus recursive query processing techniques are implemented.

3) Parallel processing methods for data stream: the stream data that push up to cloud storage and the processing algorithm is tasked with data without explicitly storing it.

The disadvantage of parallel frameworks in the cloud such as MapReduce and its variations is an unable to support complex parallel processing effectiveness. Basic algorithms of the sequential pattern may raise the scalability challenge when dealing with large data.

For problems decision of optimizing parallel data mining, a heuristic cloud bursting algorithm, Maximally Overlapped Bin packing driven Bursting (MOBB), is developed. It considers the time overlap to improve data mining parallelization. The authors [33] present Ripple, a middleware that is built on iterated MapReduce for distributed data analytics with the support of different styles of analytics in the same platform and on the same data.

Mainly, distributed processing in cloud environment is based on MapReduce. It can be carrying out, after the expansion of different type (structured, semi-structured and unstructured) data. However, on consideration of some MapReduce disadvantages, such as high communication cost, unneeded processing and lack of interaction ability in real-time processing, the methods of high-performance distributed data processing without MapReduce are required in some application related to complex processing.

In large IoT data environment, data can be defined by types, state and analysis tasks. Parallel and particle data processing framework is needed to enable the execution MapReduce pattern in dynamic cloud infrastructures, in contrast with centralized master server implementations. These re-build and execution data mining algorithm are not applicable for big data analysis system. Despite its evident merits such as scalability, fault-tolerance, ease programming, and flexibility, MapReduce has limitation in interactive or real-time processing on handling IoT data processing and is not a uniform decision for every large-scale analytical task. Its high communication cost and redundant processing is an IoT application problems.

Conclusion

The DM technique is top-of-the-agenda in the IoT concept that arises from the need to manage and analyze big sensors data. With that, DM algorithm selection for IoT is not a huge challenge itself, it mainly depends on the task and also the type of data that we are dealing with. Instead, many other issues should be resolved, they are cleaning the data; transforming all data in a unified format; struggling with missing values and/or reducing massive data sets; understanding the informational content of the data or data interestingness rate; establishing whether the data is sufficient to the purpose of DM or not.

The using of DM techniques for IoT are directs to map reduce, finding similar items. It helps to develop, control and monitor the IoT-based application in different areas.

Acknowledgment

This material is based upon work supported by the Erasmus+ ALIOT project (Reference number 573818-

EPP-1-2016-1-UK-EPPKA2-CBHEJP) focused on integration all existed and new curriculum, educational materials and tools for providing training and consultancy services in the area of IoT-based systems for different application domains and adaptation of academic programs in Ukraine and EU countries to the needs of the world labor market related to the IoT.

References

1 Jaokar, A. (2019). Data Science for Internet of Things (IoT): Ten Differences from Traditional Data Science. [online] Kdnuggets. com. Available at: https://www.kdnuggets.com/2016/09/data-science-iot-10-differences.html [Accessed 22 Dec. 2019].

2 Chen, F., Deng, P., Wan, J., Zhang, D., Vasilakos, A.V. and Rong, X., 2015. Data mining for the internet of things: literature review and challenges. International Journal of Distributed Sensor Networks, 11(8), p.431047. Available at: https://journals.sagepub.com/doi/full/10.1155/2015/431047 [Accessed 22 Dec. 2019].

3 A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, and C. A. C. Coello, A survey of multiobjective evolutionary algorithms for data mining: part I, IEEE Transactions on Evolutionary Computation, vol.18, no.1, pp.4–19, 2014.

4 Bhuiyan, M.Z.A. and Wu, J., 2016, October. Event detection through differential pattern mining in internet of things. In Mobile Ad Hoc and Sensor Systems (MASS), 2016 IEEE 13th International Conference on (pp. 109-117). IEEE.

5 Lee, D. and Lee, H., 2018. IoT service classification and clustering for integration of IoT service platforms. The Journal of Supercomputing, pp.1-17.

6 Viswanathan, H. et al (2015) Uncertainty-aware autonomic resource provisioning for mobile cloud computing. IEEE Trans Parallel Distrib Syst 26(8):2363–2372.

7 Chen, H. et al (2016) Uncertainty-aware real-time workflow scheduling in the cloud. In: 2016 IEEE 9th; International Conference on Cloud Computing (CLOUD). IEEE, pp 577–584.

8 Jamshidi, P., Pahl, C., Mendonça, NC (2016) Managing uncertainty in autonomic cloud elasticity controllers. IEEE Cloud Comput 3(3):50–60.

9 Gupta P., Gupta R. Data Mining Framework for IoT Applications. International Journal of Computer Applications (0975 – 8887) Volume 174 – No.2, September 2017.

10 Tsai, C.-W., Lai, C-F., Chiang, M.-C., Yang, L.T. Data Mining for Internet of Things: A Survey. IEEE Communications Surveys & Tutorials, Vol. 16, No. 1, first quarter 2014. pp. 77-97.

11 Mahdavinejad, M.S., Rezvan, M., Barekatain, M., Adibi, P., Barnaghi, P. and Sheth, A.P., 2018. Machine learning for Internet of Things data analysis: A survey. Digital Communications and Networks, 4(3), pp.161-175.

12 Mahdavinejad, M.S., Rezvan, M., Barekatain, M., Adibi, P., Barnaghi, P. and Sheth, A.P., 2018. Machine learning for Internet of Things data analysis: A survey. Digital Communications and Networks, 4(3), pp.161-175.

13 C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

14 Amini, A., Saboohi, H., Ying Wah, T. and Herawan, T., 2014. A fast density-based clustering algorithm for real-time internet of things stream. The Scientific World Journal, 2014.

15 Yankine, I., 2017. Unsupervised clustering of IoT signals through feature extraction and self organizing maps.

16 Boukerche, R.W. Pazzi, and R.B. Araujo, A fast and reliable protocol for wireless sensor networks in critical conditions monitoring applications, Proc. 7th ACM Int. Symp. MSWiM., pp. 157-164, 2004

17 A. Boukerche and S. Samarah, Novel Algorithm for Mining Association Rules in Wireless Ad-hoc Sensor Networks, IEEE Tran, on Par. & Dis. Sys., pp. 865-877, 2008.

18 S.K. Tanbeer, C.F. Ahmed and B.S. Jeong, An Efficient SinglePass Algorithm for Mining Association Rules from Wireless Sensor Networks, IETE Technical Review, Vol. 26, 2009.

19 M. Rashid, I. Gondal, and J. Kamruzzaman, Mining associated patterns from wireless sensor networks, IEEE Transaction on Computers, vol. 64, no. 7, pp. 1998–2011, 2014.

20 Y.K. Lee, W.Y. Kim, Y.D. Cai and J. Han, CoMine: Efficient Mining of Correlated Patterns, Proc. on ICDM, 2003.

21 Kaur, H., Wasan, S.K., Al-Hegami, A.S., Bhatnagar, V.: A unified approach for discovery of interesting association rules in medical databases. In: Perner, P. (ed.) ICDM 2006. LNCS, vol. 4065, pp. 53–63. Springer, Heidelberg (2006). doi:10.1007/11790853_5.

22 T. Hu, H. Chen, L. Huang, and X. Zhu, A survey of mass data mining based on cloud-computing, in Proc. Anti-Counterfeiting, Secur. Identificat., Aug. 2012, pp. 1–4.

23 Marjani, M., Nasaruddin, F., Gani, A., Karim, A., Hashem, I.A.T., Siddiqa, A. and Yaqoob, I., 2017. Big IoT data analytics: architecture, opportunities, and open research challenges. IEEE Access, 5, pp.5247-5261.

24 A. Gani, A survey on indexing techniques for big data: Taxonomy and performance evaluation, Knowl. Inf. Syst., vol. 46, no. 2, pp. 241–284, 2016.

25 Ding, G., Wang, L. and Wu, Q., 2013. Big data analytics in future internet of things. arXiv preprint arXiv:1311.4112.

26 D. Mari and S. Kotz, Correlation and Dependence. London, U.K.: Imperial College Press, 2001

27 S. G. Iyengar, P. K. Varshney, and T. Damarla, “A parametric copula-based framework for hypothesis testing using heterogeneous data,” IEEE Transactions on Signal Processing, vol. 59, no. 5, May 2011.

28 Shi Nash, A. and Hardoon, D.R., 2017. Data analytics and predictive analytics in the era of big data. Internet of Things and Data Analytics Handbook, pp.329-345.

29 Cai, H., Xu, B., Jiang, L., & Vasilakos, A. V. (2016). IoT-based Big Data Storage Systems in Cloud Computing: Perspectives and Challenges. IEEE Internet of Things Journal, 1–1. doi:10.1109/jiot.2016.2619369

30 S. Blanas, J. M. Patel, V. Ercegovac, J. Rao, E. J. Shekita, and Y. Tian, A comparison of join algorithms for log processing in mapreduce, in Proceedings of the 2010 ACM SIGMOD International Conference on Management of data. ACM, 2010, pp. 975–986.

31 K. Lu, M. Sun, C. Li, H. Zhuang, J. Zhou, and X. Zhou, Wave: Trigger based synchronous data process system, in Cluster, Cloud and Grid Computing (CCGrid), 2014 14th IEEE/ACM International Symposium on. IEEE, 2014, pp. 540–541.

32 A. Lugowski, S. Kamil, A. Buluc,, S. Williams, E. Duriakova, L. Oliker, A. Fox, and J. R. Gilbert, Parallel processing of filtered queries in attributed semantic graphs, Journal of Parallel and Distributed Computing, vol. 79, pp. 115–131, 2015.

33 K. Lee, L. Liu, Y. Tang, Q. Zhang, and Y. Zhou, Efficient and customizable data partitioning framework for distributed big rdf data processing in the cloud. in IEEE CLOUD, 2013, pp. 327–334.

34 Elmisery, A. M., Sertovic, M., & Gupta, B. B. (2018). Cognitive Privacy Middleware for Deep Learning Mashup in Environmental IoT. IEEE Access, 6, 8029–8041. doi:10.1109/access.2017.2787422

35 Canzian, L., & Der Schaar, M. V. (2015). Real-time stream mining: online knowledge extraction using classifier networks. IEEE Network, 29(5), 10–16. doi:10.1109/mnet.2015.7293299

36 Carey, M.J., Ceri, S., Bernstein, P., Dayal, U., Faloutsos, C., Freytag, J.C., Gardarin, G., Jonker, W., Krishnamurthy, V., Neimat, M.A. and Valduriez, P., 2008. Data-Centric Systems and Applications; Minos Garofalakis, Johannes Gehrke, Rajeev Rastogi (eds.) - Data Stream Management_ Processing High-Speed Data Streams (2016, Springer-Verlag Berlin Heidelberg).

Критська Я.О., Білобородова Т.О., Скарга-Бандурова І.С. Інтелектуальний аналіз даних для IoT аналітики

Інтелектуальний аналіз даних є однією з найбільш цінних технологій, що дозволяють виявляти невідомі шаблони і підвищувати ефективність технології Інтернет речей (IoT). Поточне дослідження присвячене процесам виявлення даних і знань для IoT. У цій статті ми представляємо систематичний огляд різних моделей інтелектуального аналізу даних і обговорюємо методи, що застосовуються для різних даних IoT. Проаналізовано деякі специфічні особливості даних і розглянуті алгоритми виявлення знань для даних IoT. Обговорюються проблеми і можливості для видобутку мультимодальних, гетерогенних, зашумленних, неповних і незбалансованих даних, а також масивних наборів даних в IoT.

Ключові слова: *інтелектуальний аналіз даних, Інтернет речей, IoT, виявлення знань в базі даних, KDD, масивні набори даних.*

Критская Я.А., Белобородова Т.А., Скарга-Бандурова И.С. Интеллектуальный анализ данных для IoT аналитики

Интеллектуальный анализ данных является одной из наиболее ценных технологий, позволяющих выявлять неизвестные шаблоны и повышать эффективность Интернета вещей (IoT). Текущее исследование посвящено процессам обнаружения данных и знаний для IoT. В этой статье мы представляем систематический обзор различных моделей интеллектуального анализа данных и обсуждаем методы, применимые для различных данных IoT. Проанализированы некоторые специфические особенности данных и рассмотрены алгоритмы обнаружения знаний для данных IoT. Обсуждаются проблемы и возможности для добычи мультимодальных, гетерогенных, зашумленных, неполных и несбалансированных данных, а также массивных наборов данных в IoT.

Ключевые слова: *интеллектуальный анализ данных, Интернет вещей, IoT, обнаружение знаний в базе данных, KDD, массивные наборы данных.*

Критська Яна Олександрівна – асп. кафедри комп'ютерних наук та інженерії Східноукраїнського національного університету імені Володимира Даля, e-mail: kritskayana@gmail.com

Білобородова Тетяна Олександрівна – к.т.н., ст. викл. кафедри комп'ютерних наук та інженерії Східноукраїнського національного університету імені Володимира Даля, e-mail: beloborodova.t@gmail.com

Скарга-Бандурова Інна Сергіївна – д.т.н., професор, зав. кафедри комп'ютерних наук та інженерії Східноукраїнського національного університету імені Володимира Даля, e-mail: skarga-bandurova@snu.edu.ua

Стаття подана 19.08.2019.