BULGARIAN ACADEMY OF SCIENCES
INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES

Kristina Ivanova Dineva

INTEGRATION OF HETEROGENEOUS DATA FROM DISTRIBUTED IoT DEVICES

A B S T R A C T

of thesis for awarding educational and scientific degree PhD

Scientific Specialty: „Informatics“
Professional area: „4.6. Computer sciences“

Scientific supervisor: Assoc. Prof. Tatiana Atanasova, PhD

Sofia, 2020

The thesis contains:

- 166 pages,
- 55 figures,
- 16 tables
- 175 bibliography sources.

Introduction

The information revolution is a result of the development of modern technologies. It is based on the achievements of scientific and technological progress. The growing presence of WiFi and 5G wireless internet access is leading to rapid technology development and an increasing number of interconnected devices.

For the paradigm of "Internet of Things" (IoT), the processing of data acquired by IoT devices can be distinguished as a stand-alone academic discipline aimed at developing and upgrading methods and tools of great importance for the increasing productivity, competitiveness of production and quality of life. Networks of interconnected devices not just extract information from the environment and interact with the physical world, but also use existing Internet standards to provide data transfer services in a remote cloud environment. This results in the generation of huge amounts of data that need to be stored, processed, and presented in an easy and efficient to interpret form. Cloud services can provide a virtual infrastructure for such processes that allow the data from monitoring devices, to be processed, analysed, modelled, and visualized in specially designed platforms.

At the same time, the development of the Internet of Things raises significant challenges that make it difficult to reveal the full potential it has.

This dissertation analyses methods and tools for integration, processing and modelling of heterogeneous data obtained from distributed IoT devices. Based on that, software and hardware solutions have been developed and implemented which have significant theoretical and practical importance.

The objective and tasks of the dissertation

The aim of the dissertation is to develop a system and tools for processing, modelling and integration of heterogeneous data obtained by distributed IoT devices.

To achieve this goal, the following tasks are defined:

1. To propose a methodology for processing, modelling and integration of heterogeneous data obtained from distributed IoT devices.
2. To develop a modular hardware system architecture and to design an appropriate method for communication between distributed IoT devices.
3. To propose an architecture of a software platform and an approach for the organization of the services for the inteligent processing of heterogeneous data from an IoT system.
4. To construct valid machine learning models for experimental validation of the proposed methodology.
5. To provide possible application in intelligent agriculture of the proposed system and tools for integration of heterogeneous data obtained from IoT distributed devices.

Dissertation structure

The dissertation has five chapters.

Chapter one presents an analytical review is provided about the theoretical basis related to the problem area of the dissertation. It includes a short introduction, relevance of the topic, applications, challenges, and existing research solutions. The explanation is given for the need to build and apply a new methodology for working with heterogeneous data, which extends and improves the existing approaches.

Chapter two presents a systematic methodology for processing, modelling and integration of heterogeneous data obtained from IoT devices. A general conceptual schema of development is presented. At the next stage, the theoretical-methodological framework is described and explained in detail as a series of steps, grouped in four main stages. In the process of defining the stages, a review is performed together with a comparative analysis of the existing methods and approaches in identifying specific cases for which their application is correct.

Chapter three describes the process of architecting and developing an IoT platform. It consists of two dedicated systems for hardware and software, which are able to communicate with each other. The first section of the chapter presents the hardware architecture and an innovative method for IoT devices communication. The second section presents architectural solutions for software implementation of a server application based on microservices and implementation of a client web interface. During the development of the software system a new approach was used for the organization of the services responsible for the intelligent data processing and exchange in the IoT system.

Chapter four presents experimental implementation and validation of the developed methodology is accomplished. Two types of problems are identified, and solutions are provided for classification and regression analysis. For the purposes of the solutions, all the steps described in the methodology have been strictly followed. As a result of the performed experiment, validated models for machine learning are built, which are ready for integration in a production environment.

Chapter five presents a practical application of the developed system. The needs and benefits of its use are considered. A comparative analysis is done between the existing systems on the market. Based on this analysis, a comparative characteristic has been compiled, summarizing the usefulness of the existing systems in relation to the developed IoT system in this dissertation.

The **Conclusion** presents a summary of the results obtained from the development. Guidelines for future research and improvement have been identified. A list of scientific publications on the dissertation and noted citations are presented.

Chapter 1 – Analysis of the state of the study

1.1 Industry 4.0

Industry 4.0 is an intensive information transformation of traditional production processes in a connected data environment and includes many innovative elements (Jazdi N., 2014) that are part of production technology, such as: Internet of Things, Artificial Intelligence, Machine Learning, Machines to Machines, Big Data and more. Also, Industry 4.0 has a strong emphasis on security. This means not only security of data and communication networks, but also data protection.

1.2 Internet of things

The Internet of Things may be considered as a unified network connecting physical and virtual objects. A scenario can be described in which uniquely identifiable computing devices are embedded in a large number of distinguished objects. Internet connectivity allows them to collect, store, share and analyse data and to be controlled remotely via other devices through an Internet connection.

1.2.1 IoT Technologies

The Internet of Things technology stack can be divided into four main layers:

- Hardware - devices that represent the "things" on the Internet of Things.
- Software - the key element that makes connected devices "smart".
- Communications - the communication layer includes both physical connectivity solutions and specific protocols used in different IoT environments.
- Platform - the place where all this data is collected, managed, processed, analysed and presented in a user-friendly way.

1.3 IoT Usages

The Internet of Things finds application in various areas such as (Table. 1.1).

Table 1.1 Areas and applications of IoT (Atanasova T., 2019)

Areas	Applications
Intelligent urban environment	Smart Cities, Smart Buildings, Smart Lighting, Cultural Behaviour Monitoring, Lifestyle Monitoring, Public Safety, Drone Surveillance Systems, Portable Personal Digital Assistant;
Intelligent energy network	Intelligent energy system, Smart consumers, Smart meters, Green energy;
Smart healthcare	Health monitoring, Health prognosis, Ambient life support, Neonatological care;
Smart transport	Traffic control, Smart routing, Smart parking, Pedestrian recognition, Smart vehicles, Autonomous vehicles;
Smart house	Smart Home Appliances, Smart Furniture, Smart Thermostats, Smart Sockets, Smart Locks, Smart Home Security Systems;
Smart farming	Precision farming, Smart pest control, Smart livestock management, Smart greenhouses and stables, Smart monitoring of storage facilities;

1.4 IoT Challenges

Based on analyses performed by Juniper Research (Sorrell S., 2018), now, the greatest challenges to the successful IoT devices implementation are connectivity, speed, compatibility, large volumes of heterogeneous data, intelligent analysis, and security.

1.5 Heterogeneous data

Heterogeneous data is characterized by a wide variety of value types and formats which are often ambiguous and with poor quality due to the presence of noise and/or missing values. Integrating such disparate data is a difficult and complex process. Therefore, in this form, the data do not bring many benefits to the needs of business applications.

Heterogeneity is one of the main characteristics of the data acquired by IoT systems and it is the main reason for the problems in the following integration and analysis.

1.6 Existing solutions for integrating heterogeneous data

Data integration is steadily developing its approaches - from historically used manual and automated integration methods to clear, orderly scientific, technical, and business processes such as:

ETL (Extract Transform Load) – data integration process, which involves three main steps - retrieval, conversion and loading, used to aggregate data from multiple sources (Atwal H., 2020).

ELT (Extract Load Transform) – three steps process for data integration - extraction, loading and conversion (Anoshin D., 2020).

CDC (Change Data Capture) - a data integration process based on identifying, capturing and providing changes made to data sources (Singh J., 2019).

OSEMN (Obtain Scrub Explore Model iNterpret) – standardized and widely accepted process of organization of research in the field of Data Science for data integration (Guhr S., 2019).

1.7 Conclusions

Based on the conclusions made, it can be summarized that for the current problems of working with heterogeneous data obtained by IoT devices, it is important to make timely decisions by creating and using a unified and systematic approach, which will lead to sustainable, reliable and reproducible results.

Chapter 2 – Methodology for processing, modelling and integration of heterogeneous data

The methodology describes "the general research strategy that outlines how research should be carried out" (Howell K., 2013). In its essence, it is a system or group of methods, rules and statements used in a particular discipline.

This chapter of the dissertation proposes a methodology (Task 1) that combines approaches, methods, processes, tools, and good practices to answer the question “How to process, model and integrate heterogeneous data from distributed IoT devices”. It is applicable to all IoT systems collecting heterogeneous data that need to be persisted, processed, and visualized.

In order to get an in-depth idea of the developed methodology, Figure 2.1 presents a detailed scheme, which describes the main stages and the steps included in them.

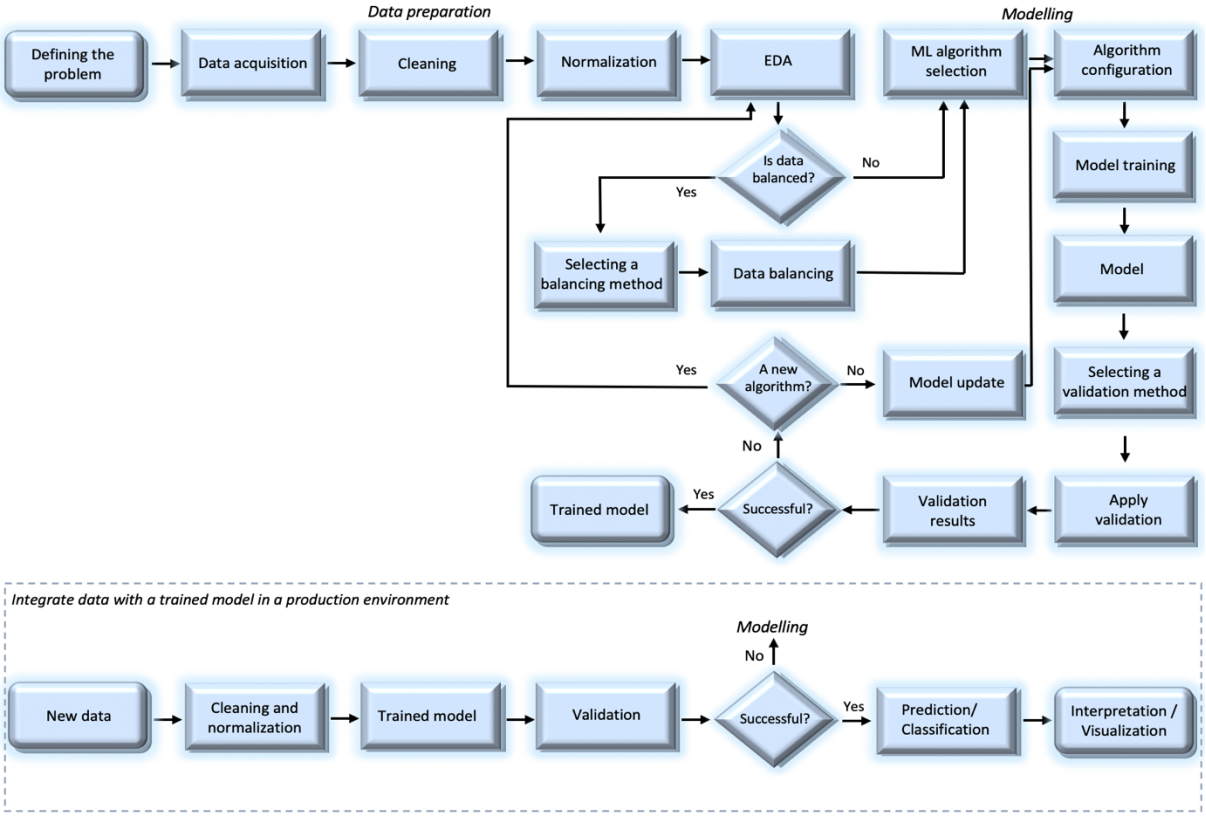

Fig. 2.1: Detailed scheme of the methodology for processing, modelling and integration of the heterogeneous data

The presented methodology consists of the following stages and accompanying steps:

Stage 1: Defining the problem

Stage 2: Data preparation:

- Extraction of heterogeneous data from IoT devices.
- Cleaning of the acquired data.
- Transformation of heterogeneous data.
- EDA.

Stage 3: Modelling:

- Machine learning algorithm selection.
- Algorithm configuration.
- Training and validation of the model.

Stage 4: Integrating data with a trained model in a production environment:

- Obtaining new data.
- Cleaning and normalization of the acquired data.
- Integrating new data into a trained model.
- Results validation.
- Data interpretation and visualization.

One of the most important features of the methodology is that it unequivocally describes not only the sequence but also the responsibilities. At each stage of the process, it is clear what is expected of it, from which one it will receive a request for a particular activity and to which one it should provide the result. The identification of existing good practices is facilitated. Achieving greater transparency of the performed activities at each stage of the process provides an opportunity to easily detect errors and quickly return to a certain stage of the process to eliminate them.

2.1 Data Preparation

2.1.1 Data acquisition

Data acquisition is the first step in the whole workflow. It decides the ultimate success of the set goal.

2.1.2 Data cleaning

When acquiring data from a work environment, there are often various reasons why there is data with a value of Null, NaN or NA. Commonly used approaches to solve this problem are to replace missing data with averages, with the most common values, or to directly remove these values.

2.1.3 Data normalization

It is necessary to normalize the data obtained by IoT systems due to the different type and range it has. The most popular methods for performing normalization are Zscore, MinMax, LogNormal и TanH (Saif S., 2017) (Samariya D., 2020).

2.1.4 Research and analysis

Exploration, detection, structuring, and analysis are operations that are extremely useful for gaining insights into the collected data. The study of a set of raw data helps to choose the best approach for conducting analytical studies (Waltenburg E., 2012).

2.2 Modelling

In the machine learning paradigm, the model refers to the mathematical expression of the model parameters along with the input characteristics for each prediction, class, and action for regression, classification, and reinforcement categories.

2.2.1 Analysis of machine learning algorithms

There are different groups of algorithms in machine learning, (Figure 2.2). The choice of a proper algorithm for solving a given problem depends mainly on the type and size of the data, as well as on its quality and quantity (Kolchakov K., 2018).

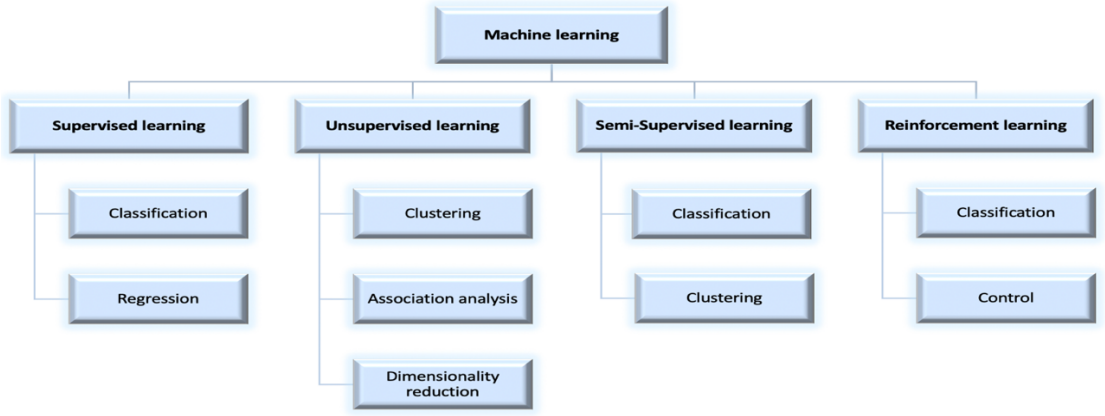

Fig 2.2: Types of machine learning algorithms

Ensemble methods

Machine learning and data science require more than just using already known and existing algorithms. An important rule is to know which algorithms can apply boosting methods (Mease D., 2007). These methods are used to build so-called ensemble models, which can help improve the accuracy of the algorithm and make the model performance more stable. This is achieved by combining weak algorithms with an ensemble method (Agarwal K., 2020), (Tewari S., 2020) and (Balabanov T., 2020).

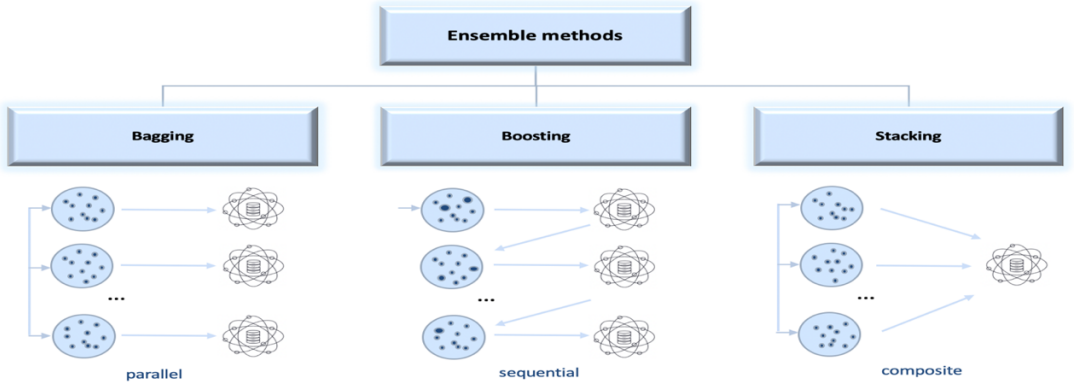

Fig 2.3: Types of ensemble methods

There are 3 methods for boosting algorithms (fig. 2.3):

Bagging - Designed to improve the stability and accuracy of machine learning algorithms.

Boosting - This method reduces bias (false positives).

Stacking - This method improves accuracy in forecasting.

2.2.2 Validation method analysis

At the Validation step the accuracy of the model is measured. It is produced by dividing the original sample into a training kit and a test kit. Training and testing methods use different approaches to split the data (Vanwinkelen G., 2012) (Balabanov T., 2018). These approaches are Train-Test Split, Cross-Validation, Early Stopping.

2.2.3 Validation of the selected model

By validating the model, a true assessment of the productivity and sustainability of the trained model is obtained. It is done through various metrics specific to different categories of training used.

The main classification metrics used to measure the effectiveness of the model are based on the confusion matrix. The most used are accuracy, precision, recall and R2 score.

The main metrics for estimating regression models are mean absolute error, mean square error, relative absolute error, relative square error, and the coefficient of determination.

2.2.4 Improving the selected model

The model that describes the training data too well usually performs overfitting. The approaches to dealing with over-adjustment of the model to the data are:

1. Cross-validation.
2. Training with more data.
3. Features selection.

2.3 Integrate data with a trained model in a production environment

After creating a successful model, it is serialized and exported to a structure that contains a classification/regression object together with a prediction function. This model is deserialized and implemented in a production environment to make predictions using new data.

Adding new data to an already trained model needs to be done under certain conditions:

- The new data must have the same variable names (columns) as those used in the training. The prediction function ignores added variables if any.
- The format and the data type must also correspond to the format of the original training data.

2.4 Conclusions

This chapter proposes a new systematized methodology for processing, modelling and integration of heterogeneous data obtained by IoT devices. It describes a logical sequence of steps, making it easy to understand and use. The proposed methods and approaches that are most suitable for its implementation are analysed depending on the type of the considered problem and the quantity and quality of the acquired heterogeneous data.

As a result, the following conclusions were drawn:

1. The methodology covers all aspects of the work process - from defining the problem and determining tasks and goals, to integrating a ready-made solution in a production environment.
2. The implementation of every stage in the methodology is done by solving several smaller tasks (steps), through which one can easily identify the problem and take timely action to solve it.
3. The methodology is designed to work with heterogeneous data and offers several methods for their normalization and integration into a unified homogeneous structure.
4. The methodology allows the creation of automated data processing pipelines.
5. The methodology can be applied by specialists in various fields where working with heterogeneous data is involved.

Chapter 3 - Modular IoT system architecture

This chapter presents the complete development process of a system for data collection, data processing and modelling. It consists of remote IoT devices (Task 2) and a distributed cloud-based software application (Task 3).

Data extraction is performed by IoT devices that are integrated into beehives. The collected data is sent to a software application, where the data is processed, analysed, and modelled, and the ready results can be accessed through dedicated user interfaces.

The monitoring of bioprocesses is an important problem and challenge for researchers and information technology specialists. Beekeeping is one of the sub-sectors of agriculture where the techniques and methods for intelligent monitoring can be applied (Beecham, 2017). The integration of information technologies in the beekeeping process can improve the knowledge of beekeepers about the behaviour of individual bee colonies. One of the current goals in the field of bee monitoring is the development of tools for continuous monitoring of bees in real-time, using automatic solutions that avoid exposing bees to additional stress or unproductive activities. The purpose of these technical means is not to replace, but rather to support the beekeeper.

3.1 Modular hardware IoT system architecture

3.1.1 System requirement

Proper identification of the needs of the system is the most important step before the planning and development phases of hardware architecture. In essence, the architecture refers to the identification of the physical components of the system and their interrelationships.

3.1.2 Hardware components - compatibility and interchangeability

Compatibility and easy interchangeability of hardware components is an important feature of the system. This is one of the main requirements for it, which is met during the design process. The system needs four main categories of hardware components - sensors, logic blocks, communication elements and power supplies.

3.1.3 Grouping and hierarchy between hardware components

The described hardware components are grouped physically and logically in a specific way developed in this dissertation. The different practical configurations of the devices in the system allow various groupings to be performed, but their logical structure remains unchanged. The following types of participants are defined in the system:

- **IoT Unit** – internal hardware device.
- **IoT Manager -** an intermediate hardware device that has the capacity to serve as a manager of several IoT Units.
- **IoT Node -** the total number of all IoT Units for which one IoT Manager is responsible.
- **IoT Edge –** an external hardware device that is designed to act as a mediator.

All IoT Nodes send the collected sensor data and status information to the IoT Edge. IoT Edge also has the ability to independently retrieve data from IoT Nodes. This is done in order to increase the stability of the system by maintaining more than one communication channels between its participants. In turn, IoT Edge, based on user-defined rules, sends the collected data to the Cloud Environment (Fig. 3.1).

The IoT Edge can receive commands from the Cloud Environment, these commands can be addressed directly to it, or a specific IoT Node. In this case, Edge forwards the command to the specific IoT Node.

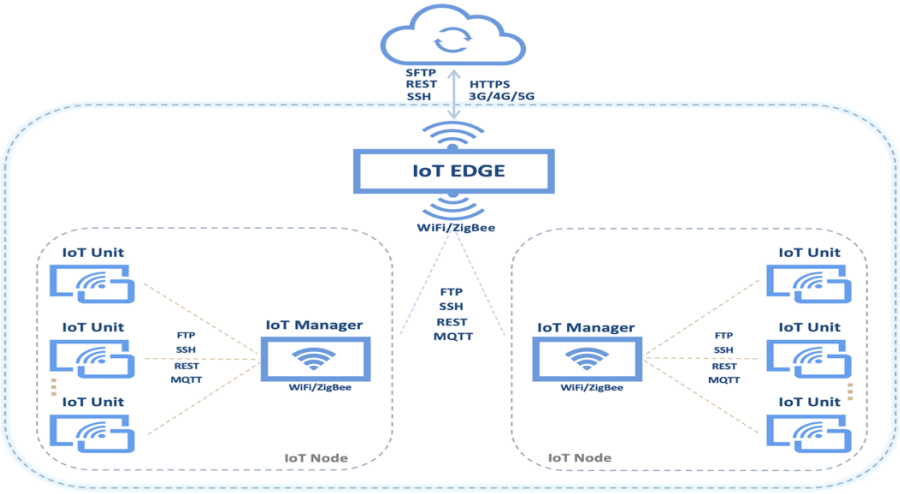

Fig 3.1: Modular IoT system architecture

3.1.4 Communication

Communication in the hardware system is based on wireless connectivity (WiFi). It is a technology that uses radio signals to transmit information between devices that support it.

The system is characterized by a centralized topology in the centre of which is a router, which also acts as a Default Gateway. In such a topology, it connects the system network to external networks and the Internet. For the better organization of the devices and semantic clarity of the logical connection between each device with its IP address, a new method for communication between the devices is proposed that is based on IP addressing segmented in specific ranges (Figure 3.2).

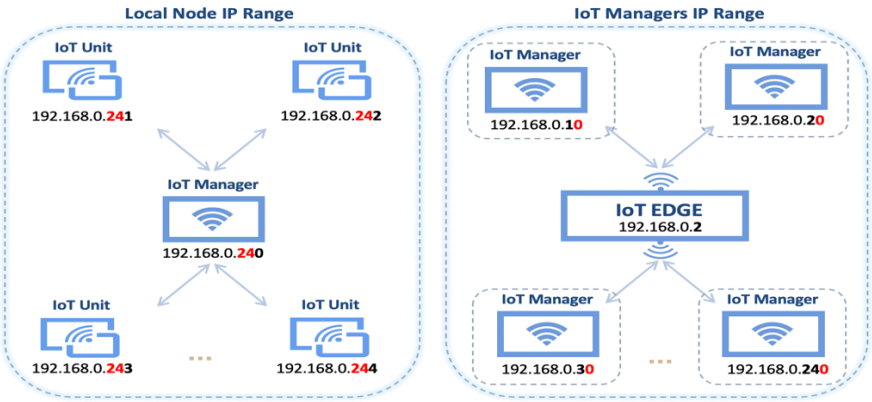

Fig 3.2: IP addressing in IoT system

The created system network and the imposed convention for IP addressing determine the following directions of communications that take place in the system (Fig. 3.3):

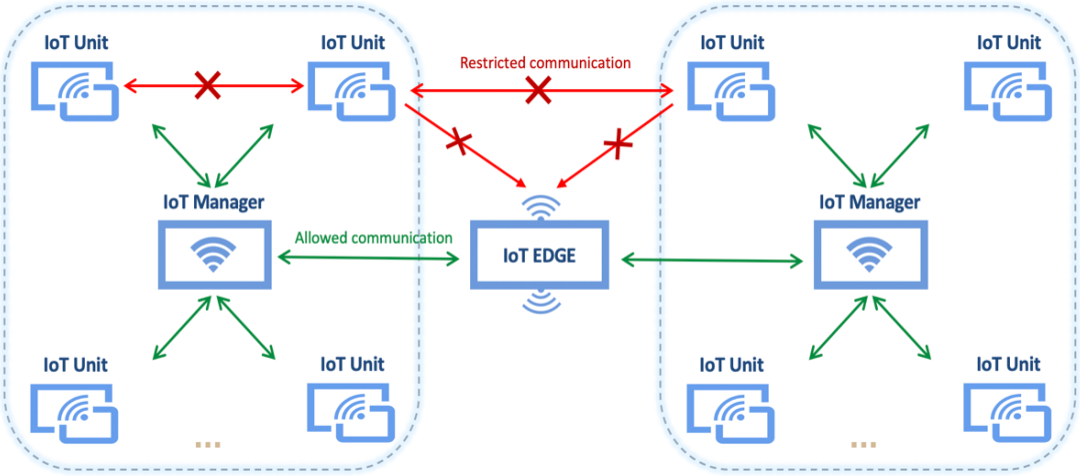

Fig 3.3: Communication in IoT system

On the base of hierarchical IP addressing a new method of communication between IoT devices has been proposed. The clear distinction of communication in the hardware system from that of the system in the Cloud Environment suggested in the dissertation is important because in case of need for changes in either of them, it will not affect the other.

3.2 Software system architecture

The established Agile procedure was followed during the development of the software system in the dissertation. Agile is a flexible approach in which software system development is based on iterations in which the execution of the steps is repeated sequentially (Cohn, 2019), (Manel D., 2019).

3.2.1 Server architecture

Defining the requirements

The main requirements to the developed server architecture can be defined as Services, Support and Flexibility. The expected supported functionalities of the system are user support, data handling, modelling, end-device management, and logging.

Architecture and Design

Design refers to the creation of software architecture. It in turn determines the overall structure of the software system, software components, properties, and relationships between them. Defining the functional requirements and the technological stack for system development leads to conclusion that the most suitable architecture is the one based on Microservices - MSA. It allows to construct and maintenance of distributed software system, work with various databases and create stand-alone user interfaces that can communicate with the server part. A new approach for organization of services for the intelligent data processing has been applied which includes the implementation of the following microservices (Fig. 3.4):

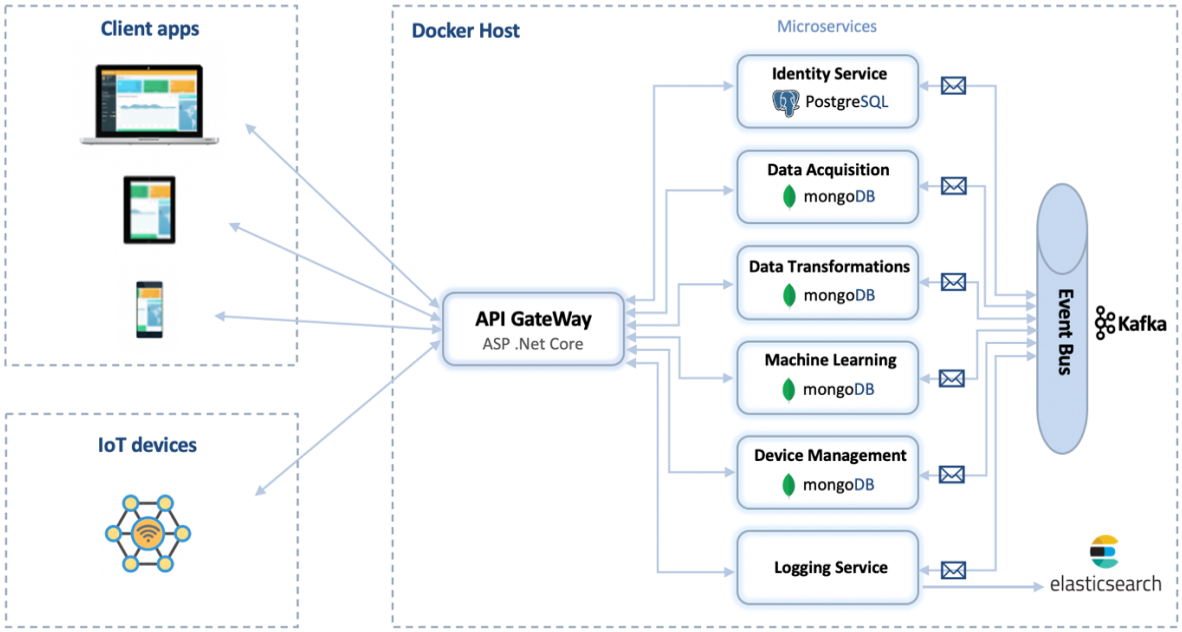

Fig 3.4: Microservice software architecture

- ***Identity Service*** is responsible for all operations related to the user.
- ***Data Acquisition*** is responsible for all data pipelines (a set of all steps taken in the process of sustainable data transfer between remote physically different locations) through which the system collects data from remote IoT devices.
- ***Data Transformations*** is responsible for all transformations on the newly received raw data until they acquire the necessary type, suitable for analysis and application of machine learning models on them.
- ***Machine Learning Service*** is responsible for applying ready-made, trained machine learning models to predict values and probable events.
- ***Device Management Service*** is responsible for transmitting commands to the user's IoT devices.
- ***Logging Service*** is responsible for the collection and analysis of events that have occurred during the operation of the system.
- ***API Gateway Service*** is used to separate user interfaces from the system.
- ***Event Bus module*** is responsible for the transmission of messages to the required recipient (Alekseev I., 2015). The Event Bus used in the system is based on Apache Kafka (Hesse G., 2020)

Development

The developing of microservices begins with specifying three main aspects: defining a bounded context, defining the types of communications and the consistency of the data.

3.2.2 User interface

The interface is a point of interaction between the user and the system. It is important for the visualization of the collected and analysed sensory data from IoT devices and how they are presented to the user.

Defining the requirements

The main requirements during the development of the user interface in the dissertation are:

- Covering user requirements.
- Fast loading and regeneration.
- Design for compatibility with various devices and browsers.
- Well-organized and meaningfully arranged content.
- Intuitive and understandable navigation structure.
- Flexibility in adding new functionalities.
- Stability and continuity of work.
- Security of user data.
- Easy to maintain and update.

The user interface of the system developed in the dissertation is thematically divided into two parts:

- Landing page
- Control panel

All parts support different functionalities that are consistent and complementary.

Development

The user interface in this dissertation was developed using Angular framework, which was created specifically for building client SPA applications (Japikse P., 2020). They are called SPA (Single-Page Application) because the entire content of the application is developed in the form of stand-alone components that load dynamically on a single web page, giving the impression that the application can actually navigate between more than one page (Fig. 3.5).

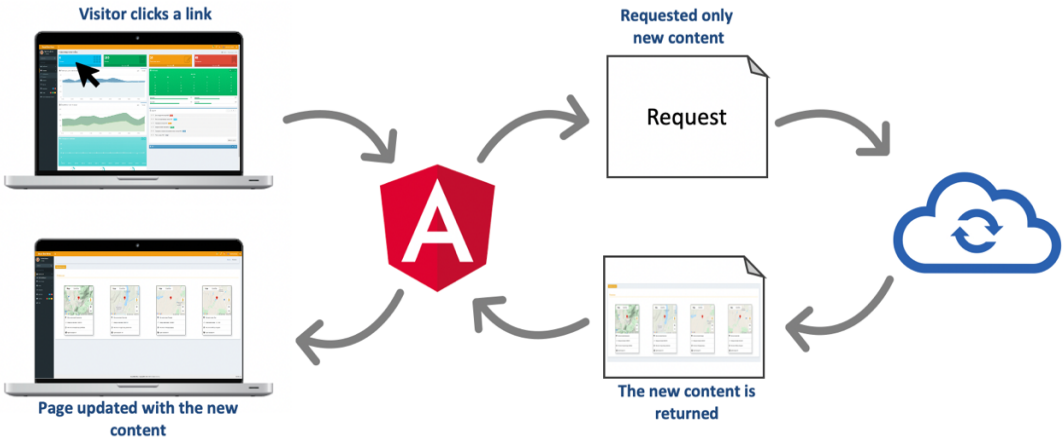

Fig 3.5: Angular process

SPA applications are characterized by a reduced amount of dynamic page refreshes when receiving data. This is achieved by using AJAX communication with the server. In fig.

3.6 "Dashboard" shows the web content that is accessed in the application via the navigation address "https://smartbeehives/dashboard". It is formed by calling and loading the components.

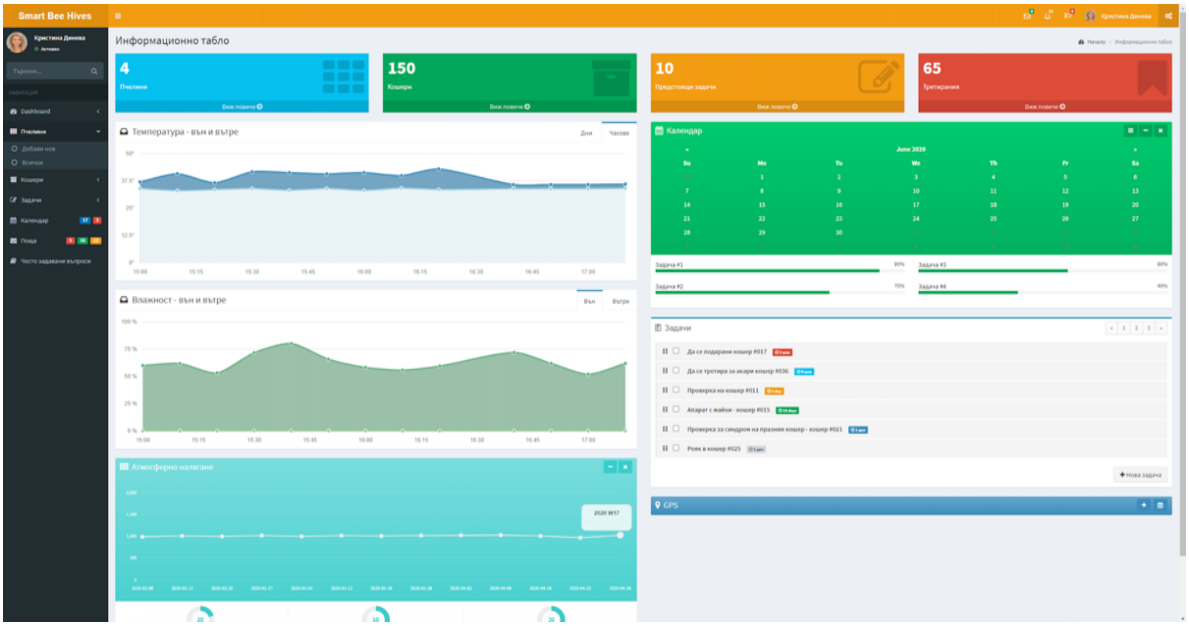

Fig 3.6: User interface - dashboard

When the user navigates to access new content, a request is sent to the server to update the relevant sections of the page. The server returns to the client only the new content in the form of new components and refreshes the page with them without reloading the existing elements (Fig. 3.7), which remain unchanged.

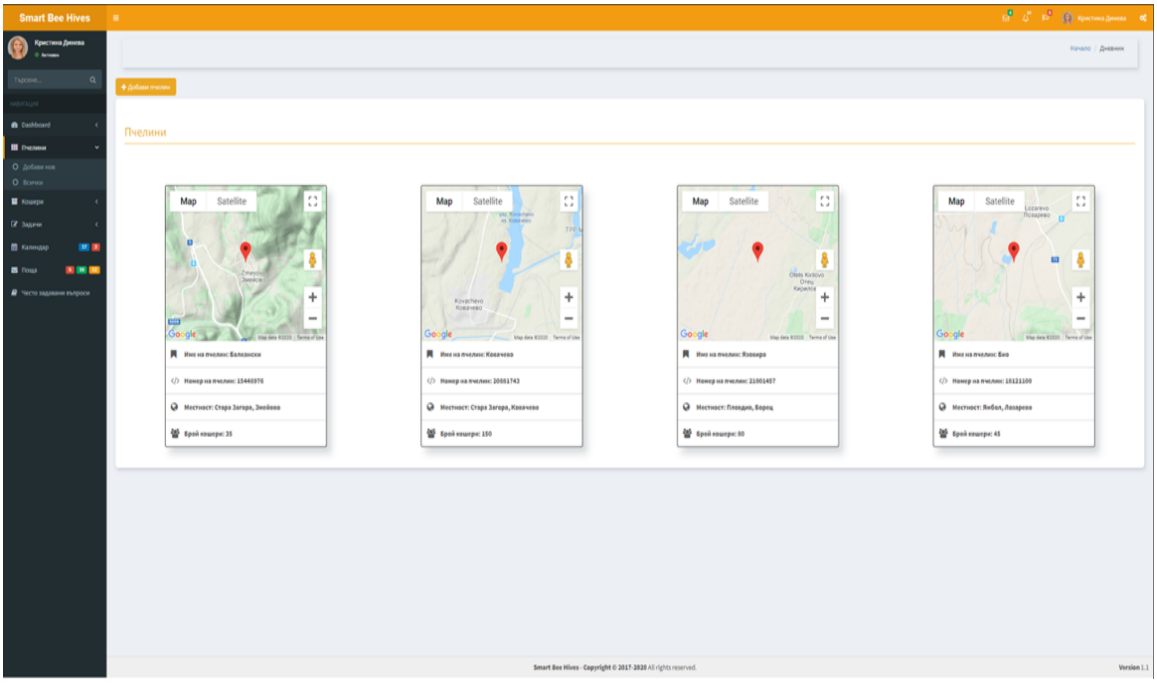

Fig 3.7: User interface - apiaries

This principle of operation provides speed in loading and updating information content and easy addition of new functionalities - by creating new components or entire modules.

3.3 Conclusion

This chapter proposes an innovative system built of IoT devices, a cloud server part and a user web interface that interact and exchange data with each other. The developed IoT hardware architecture and the proposed and implemented innovative method for communication between IoT devices, based on hierarchical IP addressing, provide significant advantages to the system.:

1. Effective work with different types of hardware components and devices, easy replacement of existing ones or adding new ones when you need to change the scale or functionality of the system.
2. Support for different communication channels from IoT devices, which allows working with different hardware components depending on the specific needs of the user, the specifics of the terrain and the number of devices used.
3. Much of the logic performed by the devices can be controlled, changed, and updated remotely through integrated software approaches.
4. Dealing with higher power consumption in the system using hardware devices for intelligent power management and a built-in real-time clock.
5. The hardware system has only one access point, which makes it a high level of security.

The software architecture is of MSA type. It consists of many microservices, which are implemented by applying a new approach for organization of the services for intelligent data processing and exchange, which provides the following advantages of the system:

1. Focus on functionality, not technology. Each microservice is designed to perform certain functionality. This makes this architecture adaptable for use in many other processes and different channels depending on needs. Each microservice is responsible for a specific service, which leads to the construction of intelligent and multifunctional architecture.
2. Improved performance and speed. Microservices can run simultaneously without having to wait for responses.
3. Easy maintenance, flexibility, and scalability of the whole system. Each microservice can be managed independently. If necessary and future development of the system, new independent microservices can be easily added.
4. Autonomy and multifunctionality. The microservice architecture provides independence and automatic operation of microservices.

The user interface is built as a SPA application, which is built from a series of components. This type of architecture has the following advantages:

1. High speed and flexibility are achieved by loading only the required content (component) and not the entire page. Unchanged content is cached.
2. A high level of code reusability is achieved - once created, components can be used in different places in the application.
3. The existing interface can be easily upgraded and enriched with additional functionalities, creating new components that will be called if necessary and visualized in the main component.

Chapter 4 - Experimental results

In this chapter the main goal is to apply and validate (Task 4) the methodology proposed in the Chapter 2 for processing, modelling and integration of heterogeneous data obtained from distributed IoT devices. It goes through all the necessary steps from the stages "Data preparation" and "Modelling" to obtain valid and effective trained models, which are integrated into the MSA software platform developed in this dissertation.

4.1 Application of the developed methodology

The sequence of actions is an important part of the research process so that the experiments can be repeated and improved in the future if necessary. For the correct organization of the whole process from loading the base, through cleaning, normalization, and division of the data, to application, training and evaluation of the models, the developed methodology is followed.

After data processing, it is essential to define the most appropriate algorithm for machine learning. This is a complex process that depends on the size, quality and nature of the data collected. The workflow to achieve the objectives of the experiment is performed in a cloud computing environment - Microsoft Machine Learning Azure Studio (Abraham T., 2018), (Azure, 2020).

4.2 Machine learning algorithm selection

4.2.1 Classification

To obtain a valid classification model for machine learning, algorithms for classification on heterogeneous data from IoT devices are applied. The data were obtained from a data centre for the study of honeybees “The Data Centre for Honeybee Research”. The models are trained to predict one of a totals of two classes: “0” - the bee family does not have enough conditions to survive or “1” - the bee family has enough conditions to survive. The training is performed based on certain parameters - internal and external temperature in the hives and humidity percentage, i.e. a binomial classification will be performed.

Training and testing of models

After loading the database, selecting the necessary columns, cleaning and normalizing the data, they are divided into two parts 70:30. 70% of the data is used to train the models and 30% is used for testing. Stratification is applied when splitting the data into parts.

We apply four types of algorithms (Fig. 4.1), selected after a comparative analysis in Chapter 2, which are:

1. Two-class Boosted Decision Tree,
2. Two-class Bayes Point Machine,
3. Two-class Logistic Regression,
4. Two-class Support Vector Machine.

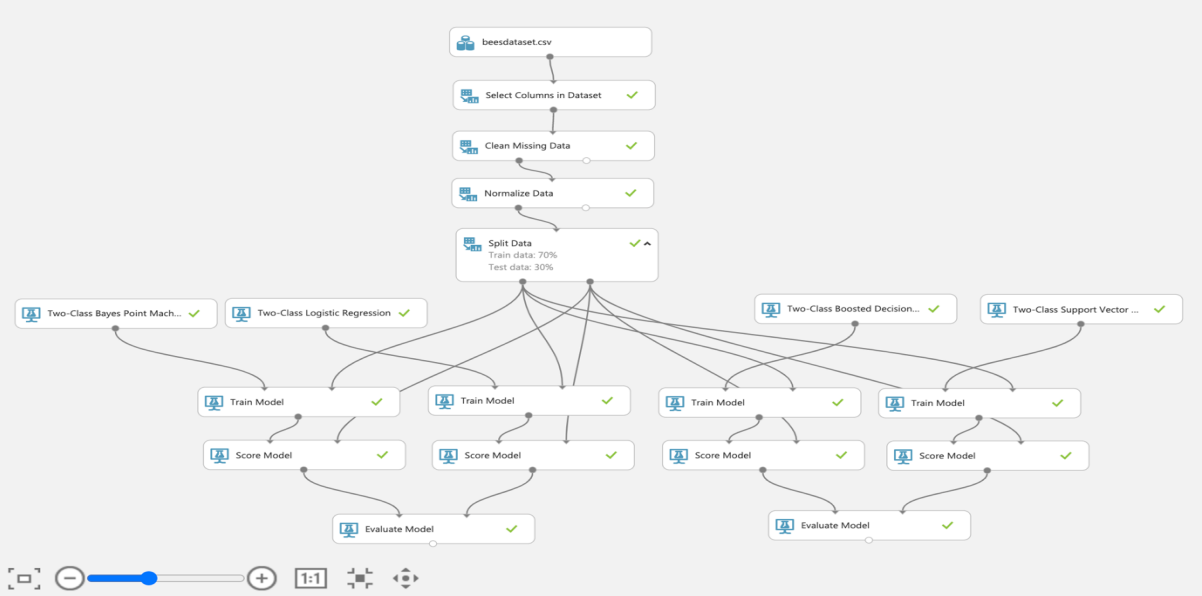

Fig 4.1 Workflow diagram

Each algorithm is the result of a modelling function that determines how the algorithm works, and hence the way the results are obtained, so it is important to make the correct configuration (setting of hyperparameters) of each algorithm.

4.2.2 Evaluate models

To determine the success of a model, it is necessary to use different metrics to evaluate it. There is no universal formula for this - it all depends on the specific data and the goals associated with them. The metrics selected for this experiment are Accuracy, Precision, Recall, and F1 Score. The calculation of the metrics for evaluation of the models is based on the constructed matrices for evaluation of the classification. After calculating the values, we obtain coefficients for each metric of each model (Table 4.1).

Table 4.1: Evaluate models

	Boosted Decision Tree	Bayes Point Machine	Logistic Regression	SVM
Accuracy	0.998	0.884	0.882	0.888
Precision	0.994	0.033	0.053	0.146
Recall	0.993	0.004	0.008	0.017
F1 Score	0.994	0.007	0.014	0.030

The comparison table contains the calculated coefficients for each algorithm. After training and testing the models and reviewing the metrics obtained from them to evaluate their effectiveness, the model Two-Class Boosted Decision Tree stands out as the best classifier compared to the others. The high success rate of this model, combined with a high score of sensitivity and precision, requires verification of the reliability of the results obtained by applying approaches to overcome the presence of unreliable results of overfitting.

Approaches to overcoming the overfitting of the model

Model overfitting is a modelling error that occurs when a function fits too tightly to the data set. To avoid over-adjusting the model, three main approaches are used:

- Data balancing.
- Feature selection.
- Cross-validation.

Data balancing

The chosen model is the Two-Class Boosted Decision Tree. The choice of data balancing technique must be taken into account. This model is sensitive to the proportions of the different classes and tends to prefer the class with the largest share of observations (also called the majority). This often leads to misleading model accuracy. Therefore, a technique was chosen to synthesize new minority cases – SMOTE (Synthetic Minority Oversampling Technique).

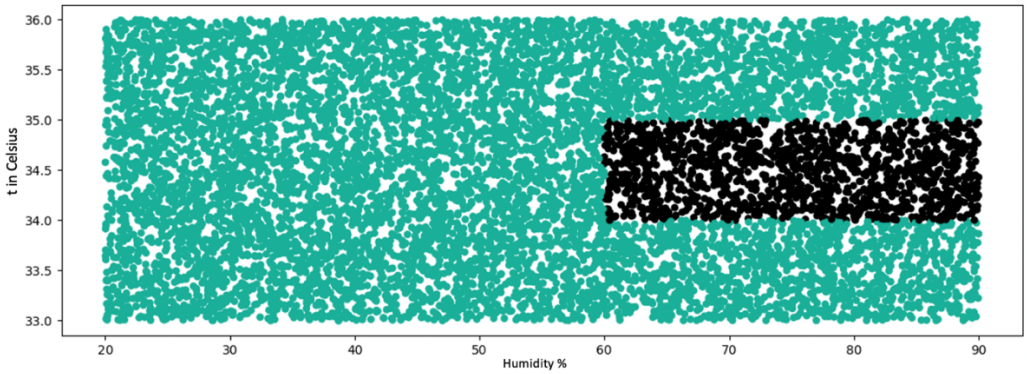

Fig 4.2: Row data distribution before applying SMOTE model

The work process for balancing the data begins with the cleaning of invalid or missing values, after which the SMOTE algorithm is applied.

After execution, a set of data is formed, which contains the original samples plus an additional number of synthetic samples from the minority, which are created to preserve the distribution of the minority class (Fig. 4.3). The number of data has been increased without new data being obtained, which would affect the extraction of information from the minority class.

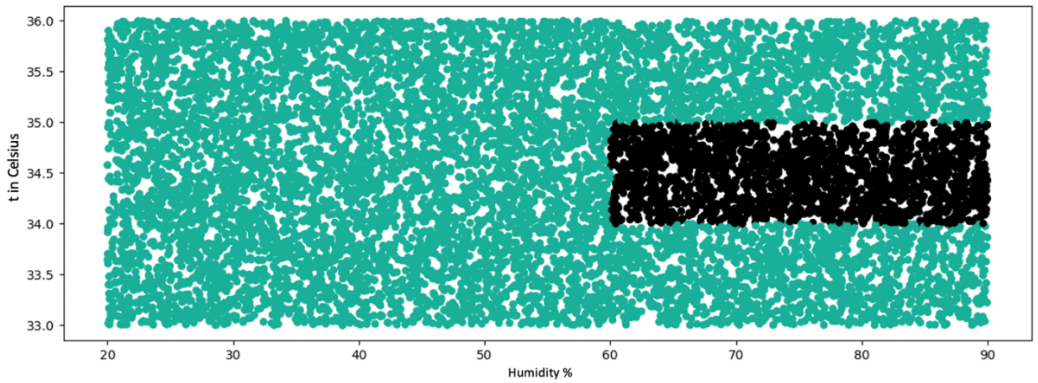

Fig 4.3: Row data distribution after applying SMOTE model

After applying the algorithm, there is an increase in the minority class from 10.39% to 37.64%, which is equivalent to 27.25% more generated copies of existing data. This, on the one hand, balances the data and, on the other hand, increases the amount of data, which obliges the model to generalize them to make progress. This leads to increased accuracy of the model while reducing the chance of overfitting.

Feature selection

Choosing the right variables improves the model and prevents problems such as:

- Overfitting.
- Inappropriate variables.
- Presence of multicollinearity. In multicollinearity, the presence of two highly correlated variables causes inaccurate calculations of other variables.

For these reasons, the following algorithms are used to correctly select a method that determines the utility of variables: Pearson correlation, Kendall correlation, and Chi-Squared.

Table 4.2: Results feature selection

	Temperature - inside	**Humidity**	**Temperature - outside**	**Accuracy**
Pearson	0,013381	0,569841	0,017359	0,792
Kendall	0,009175	0,458461	0,009175	0,800
Chi-Squared	**0,156944**	**0,12615**	**0,000597**	**0,863**

After applying the three types of variable selection algorithms to the data in combination with the selected Boosted Decision Tree algorithm for data classification, it can be seen from Table 4.2 that the application of the Chi Squared algorithm performs the best selection of variables over the others. Using Chi-Squared gives 86% accuracy compared to 79% accuracy for Pearson and 80% for Kendall.

Cross-validation

Cross-validation is an important technique in machine learning to assess both the variability of a data set and the reliability of each model trained with that data.

A total of 10 models were generated during the cross-check, and each model was trained using 8/10 of the data and tested against 2/10 of them. When the construction and evaluation process is complete for all parts, a set of performance indicators is generated, and results are noted for each fold.

Training and testing of the selected model

It is important to re-train and test the chosen algorithm in order to check the effectiveness of the applied approaches to overfitting.

The whole work process consists of eight main steps.

In the first step, the already balanced data is loaded. The new database contains 27.25% of newly generated data. After selecting the work columns, the data in them is cleaned and normalized. By applying the Chi-Squared algorithm, the weights of the variables are determined according to the degree of their influence on the final result. The Boosted Decision

Tree algorithm is then trained and tested using cross-validation. The metrics obtained from cross-validation show the success of the model to generalize the data (Table 4.3).

Table 4.3: Metrics obtained after training and testing of Boosted Decision Tree

	Boosted Decision Tree
Accuracy	0.863
Precision	0.737
Recall	0.989
F1 Score	0.845

The distribution of the results is visualized using three types of curves:

ROC curve – a graph showing the effectiveness of the classification model (Fig. 4.4a).

AUC (accuracy) – provides a generalized security measure in all possible classification thresholds (Fig.4.4b). The obtained metrics show that the accuracy of the model is 0.737 and the sensitivity is 0.989. After their calculation, the obtained coefficient is 0.902, which shows that the selected algorithm is balanced and has a high success rate.

Lift curve – a measure of the effectiveness of a classification model, calculated as the ratio between the results obtained with and without the forecast model (Fig. 4.4c).

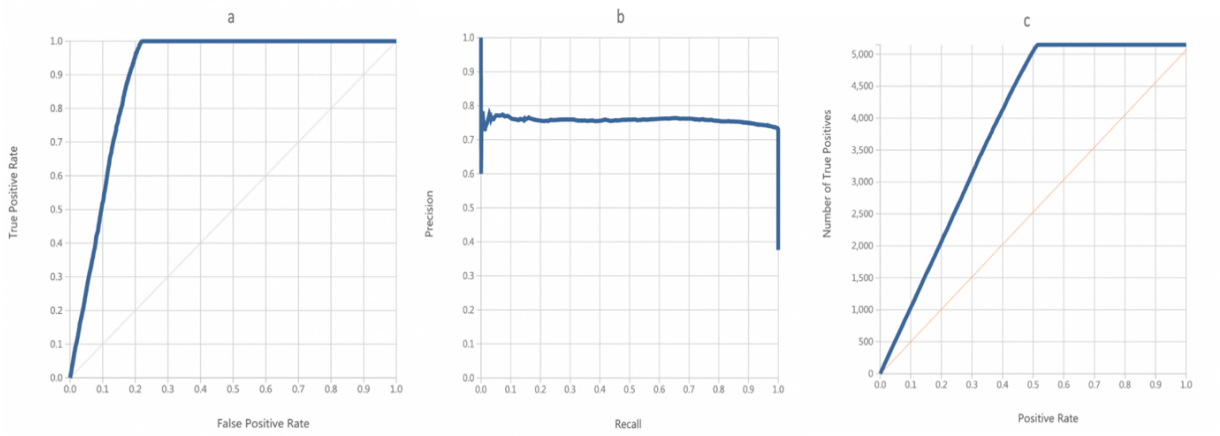

Figure 4.4 Visual distribution of results (ROC-AUC-LIFT)

Thanks to the visual presentation of the results through different types of diagrams, the effectiveness of the model used can be easily established before it is applied in a production environment.

4.2.3 Regression

Regression analysis is a set of statistical methods for estimating the nature of the relationships between variables.

Application of methods for feature selection

To achieve high accuracy, noise reduction and training time, five different methods were applied for variable selection. As a result (Table 4.4), it was found that the most appropriate method for the study was Pearson's correlation, which best describes the degree of influence of the variables. It shows that the variables are quantitative, have a normal distribution and the relationship between them is linear.

Table 4.4: Comparison of the results obtained when applying different methods for selection of variables.

	Temperature	Humidity	Atmospheric pressure	Wind - speed	Time range
Pearson	**0,58**	**0,28**	**0,44**	**0,09**	**0,01**
Kendall	0,33	0,18	0,45	0,029	0,001
Chi Squared	146,86	97,07	235,20	40,84	2,29
Fisher	2,45	1,84	1,06	0,92	0,91
Spearman	0,47	0,26	0,61	0,038	0,002

Working process

After selection of columns for work from the database, normalization of the data, clearing of the missing data and choice of variables, training of four models is performed - Linear Regression, Boosted Decision Tree Regression, Bayesian Linear Regression and Decision Forest Regression (fig. 4.5).

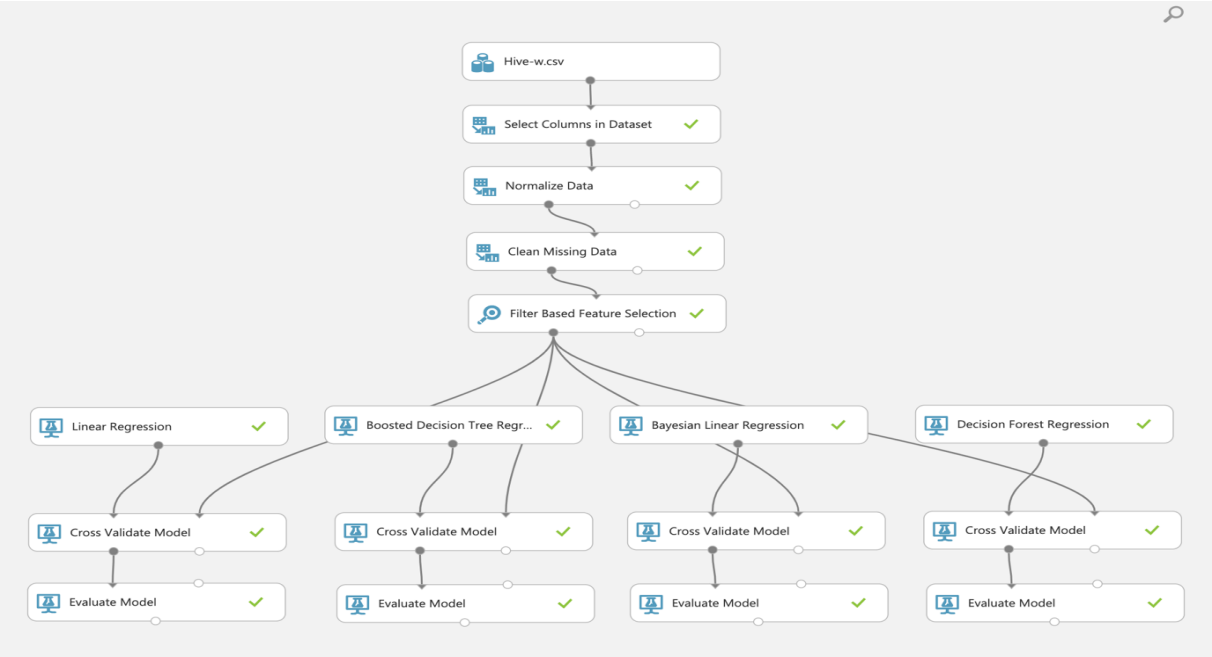

Fig 4.5: Workflow diagram

Results of work performed

The results obtained after training and testing of the chosen models are the determining factor for the correct selection of model. Table 4.5 presents the metrics obtained after training and testing of four regression models.

Table 4.5: Comparison of metrics obtained from the applied four algorithms

	Linear Regression	Boosted Decision Tree Regression	Bayesian Linear Regression	Decision Forest Regression
Mean Absolute Error	**0.031631**	0.100482	0.066806	0.087543
Root Mean Squared Error	**0.092386**	0.145703	0.098254	0.138833

Relative Absolute Error	**0.229573**	0.729287	0.484872	0.635377
Relative Squared Error	**0.209455**	0.520975	0.236910	0.473003
Coefficient of Determination	**0.790545**	0.479025	0.763090	0.526997

The tabular comparison of the obtained metrics for each of the applied regression models shows that the linear regression is the model that best describes the statistical relationship between the considered variables.

4.3 Conclusion

In this chapter, the methodology for processing, modelling and integration of heterogeneous data from distributed IoT devices proposed in Chapter 2 is applied and validated. The experiment was performed in Microsoft's cloud computing environment - Azure Machine Learning Studio. Working schemes have been developed, containing all the necessary steps of the developed methodology, which are necessary for solving the defined problems.

Problems for classification and regression are considered and solved. In resolving the classification problem, two options were considered to cover and successfully validate all the steps of the stages "Data processing" and "Modelling" of the methodology.

1. As a result of the performed experiments, the following conclusions were made: The methodology is a flexible and systematic process that needs to be followed to achieve the correct result. Skipping a step in the work process leads to overfit or underfit.
2. The selection of one of several algorithms solving the same task depends on its quality indicators. One of the most important quality indicators of the models is considered to be the time required for their implementation, the accuracy of the obtained results, and sensitivity of the data to changes. Proper configuration of each algorithm significantly increases the accuracy of the final result.
3. Data balancing in solving classification tasks plays a key role in the successful generalization of the model.
4. Determining the most directly dependent variables to the subject of the considered problem increases the accuracy and precision of the model, which leads to the best overall results.

For the IoT system to be independent of Azure cloud data processing and modelling services, a solution has been developed to validate the proposed methodology using Python programming language. The solution follows all the steps presented in the methodology. Machine learning models (for classification and regression) and their correct configuration are stored in Pickle file format.

Chapter 5 – Practical application of the developed IoT system

This chapter of the dissertation reflects one of the many practical applications of the proposed modular IoT system. The importance of integrating a monitoring system for the sector, the benefits and positive effects it has on the industry are considered. A short description of the existing similar solutions on the market has been prepared, based on which a comparative analysis of these solutions has been made about the system developed in this dissertation.

To make the presented IoT system more recognizable, it is called SmartBeeHives.

The SmartBeeHives IoT system combines the seven main components that ICT covers - software, hardware, transactions, communication, data, internet access and cloud computing. Through the combination of these components, the system performs a complete process (Fig. 5.1) - collection, transmission, processing, analysis, modelling, data visualization and event prediction.

Fig 5.1. SmartBeeHives workflow

The process begins with the implementation of hardware devices in beehives, which collect heterogeneous data on the microclimate in the hive and the atmospheric conditions outside it. This data is transmitted to the cloud via the Internet. In the cloud environment, data is transformed to perform accurate calculations and obtain useful information. The information obtained is visualized in a user interface, where the user can monitor the growth of bee colonies in the hives and be notified of future occurrence or current threat.

Technology and bees

Today, beekeeping is at the forefront of agricultural activities, which are carried out widely in the world. The development of technology and the reduction of beekeeping activities due to the ecological and climatic characteristics of the world has become an important area of research (Braga A., 2020), (Beecham, 2017).

Overview of existing systems and developments

Increasingly, the latest technology exhibitions are focusing on the Internet of Things (IoT) as a technology that contributes to sustainable development and environmental protection. Many IoT and environmental projects receive funding and start their own development. These projects are related to agriculture and the protection of bee colonies. Some of them are Arnia, ApiS, Bee Smart Technology, BeeHive Scales and BuzzTech.

Price-quality matrix

The positioning of products in one of the nine quadrants of the Kotler matrix (Kotler, 2011) shows the price-quality ratio and how it is perceived by consumers. The perceived value of consumers is an important factor in building an opinion about the product (Fig. 5.2).

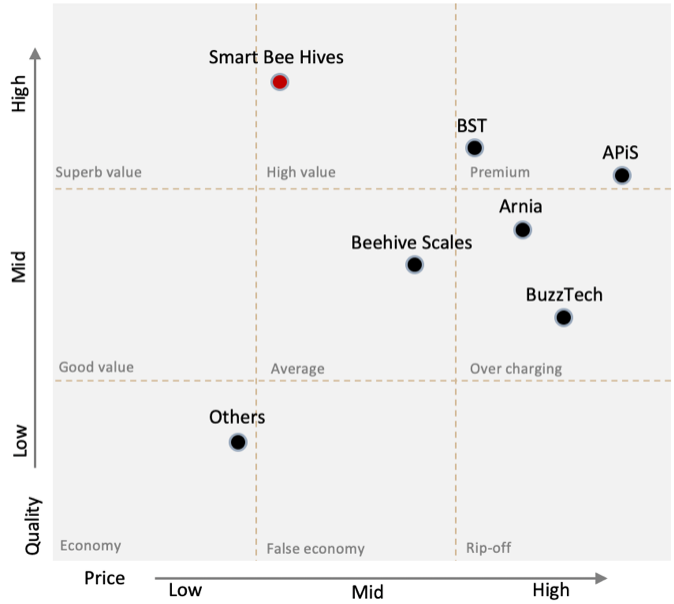

Fig. 5.2 Price-quality matrix

The golden mean for the introduction of a new product is its positioning in the quadrant "high value - average price" and high quality compared to similar products offered on the common market. SmartBeeHives finds itself in this golden environment, where quality exceeds the price. This enables consumers to get acquainted with the product and make a comparison with the products positioned in the "Premium" quadrant, where the high price corresponds to the high quality and from a consumer point of view it is a valuable investment.

Conclusion

This chapter presents a practical application of the developed system for modelling and integration of heterogeneous data provided by IoT devices. The beekeeping sector is considered as a field of application. The need to integrate new technologies in this industry and the benefits they will bring are analysed. A comparative analysis has been prepared between the existing systems on the market, based on which an economic justification of the practical application of the system in this sector has been made.

As a result, the following conclusions can be drawn:

1. New technologies enable beekeepers to make their activities more efficient and optimized, providing the opportunity for remote monitoring and early notification in the event of deviations in the normal parameters in the beehive.
2. Integrated IoT devices allow data collection for all-important internal parameters of bee colonies.
3. The existing systems are aimed at large bee farms and do not cover the needs of small producers and beekeeping enthusiasts. In contrast, the proposed modular system in this dissertation allows us to meet equally well the functional requirements of both large manufacturers and amateurs.
4. The SmartBeeHives system alone combines the seven basic components of ICT.
5. The system allows the integration of existing devices to send data to the software platform for data analysis and visualization.
6. The system is very flexible due to its variety of hardware components from which it can be built.

Conclusion and Summary of the received results

The dissertation analyses in detail the significance of the data acquired from distributed IoT devices and their role in the continuous improvement of business processes related to them. Problem areas are identified, and specific problems are defined.

A new systematized methodology is proposed, which consists of successive stages for processing, modelling and integration of heterogeneous data. The methodology has been validated in a Cloud Computing Environment, where regression and classification models for machine learning have been trained and validated, which then are used to obtain useful knowledge from the collected data and forecast future events.

A common hardware and software scalable system of a modular type for data collection and processing from distributed IoT devices has been developed and implemented. It supports efficient operation with a large number of different types of integrated hardware components and devices that cooperate using different communication protocols.

The practical results of the system are visualized in a specially built user interface, which is a point of interaction between the user and the system.

The sphere of the practical applicability of the developed system is considered and its market positioning in relation to other existing systems is analysed.

According to the work done in this dissertation and the results obtained in the course of research and discussed earlier, the following **applied scientific contributions** can be formulated:

1. A methodology for processing, modelling and integration of heterogeneous data obtained from distributed IoT devices has been developed along with performed selection for:
   - methods for working with heterogeneous data.
   - machine learning classification and regression algorithms.
   - metrics for evaluation and validation of obtained results.

2. An architecture of a modular hardware system has been developed which consists of sensors and IoT modules for control and communication. An innovative method of communication between IoT devices on the base of hierarchical IP addressing has been proposed.
3. An MSA software architecture for storage, processing and analysis of heterogeneous data has been designed and implemented. An innovative approach for the organization of the services for the intelligent data processing and exchange in the IoT system has been developed, which increases the reliability and functionality of the system and additionally provides capabilities for machine learning.
4. Machine learning models have been trained and build, which experimentally confirm the developed methodology.
5. A possible application of the developed IoT system for integration of heterogeneous data in intelligent agriculture is shown. A comparative analysis of the functional characteristics and market positioning of existing similar systems is done, through which the economic efficiency and expediency of the developed IoT system are proved.

Future development

Basic future development research on the topic of the dissertation include:

- Integrate new real-time big data platforms such as Hadoop and Spark.
- Study of processes to improve UX when using the user interface via mobile devices.
- Study of system security and analysis of potential new attack vectors. Integration of automated security systems.
- Research of new hardware components for compatibility with the existing ones. Analysis and improvement of the life cycle of the implemented hardware components.

Publications

1. **Dineva K.,** Atanasova T., Methodology for Data Processing in Modular IoT System. Distributed Computer and Communication Networks, 22-st International Conference, DCCN 2019, Springer Nature Switzerland AG 2019. V. M. Vishnevskiy et al. (Eds.): DCCN 2019, LNCS 11965, **2019**, https://doi.org/10.1007/978-3-030-36614-8_35, pp. 457 – 468, **Q2**, **SJR:0.283**
2. **Dineva K.,** Atanasova T., Integrated Systems With Embedded Sensors for Digital Agriculture. 19th International Multidisciplinary Scientific GeoConference SGEM 2019, 19, 6.1, SGEM, **2019**, ISBN:978-619-7408-88-1, ISSN:1314-2704, DOI:10.5593/sgem2019/6.1/S25.098, pp. 761-768, **Q4**, **SJR:0.232.**
3. **Dineva K.**, Atanasova T., Regression Analysis on Data Received from Modular IoT System. ESM'2019, EUROSIS-ETI, **2019**, pp. 114-120, **Scopus.**
4. **Dineva K.**, Atanasova T., Security in IoT Systems, *Proceedings 19th International Multidisciplinary Scientific Geoconference SGEM,* **2019***,* Vol. 19, Informatics, Geoinformatics and Remote Sensing, Issue 2.1, ISBN 978-619-7408-79-9, ISSN 1314-2704, DOI:10.5593/sgem2019/2.1, pp. 576-577, **Q4**, **SJR:0.232.**
5. **Dineva K.**, Atanasova T., ICT-based Beekeeping using IoT and Machine Learning, *Distributed Computer and Communication Networks,* 21-st International Conference, DCCN 2018, Revised Selected Papers, Vladimir Vishnevskiy, Dmitry Kozyrev (Eds.), Springer, Communications in Computer and Information Science (CCIS). 919, Springer, **2018**, ISBN:978-3-319-99446-8, ISSN:1865-0929, DOI: https://doi.org/10.1007/978-3-319-99447-5, pp. 132-143, **Q3**, **SJR:0.188**
6. **Dineva K.**, Atanasova T., Подходи и методи за анализ и обработка на данните в мониторингова система за пчелни кошери, Годишник на департамент „Телекомуникации", NBU, 2018, ISSN: 2534-854 X (online) No: 5, pp. 37-46.
7. **Dineva K.,** Atanasova T., Applying Machine learning against beehives Dataset. *18-th International Multidisciplinary Scientific Geoconference - SGEM 2018,* 18, 6.2, SGEM 2018, 2018, ISBN:978-619-7408-51-5, ISSN:1314-2704, DOI:10.5593/sgem2018/6.2, 35-42, **Q4**, **SJR:0.232.**
8. **Dineva K.,** Atanasova T., OSEMN process for working over data acquired by IoT devices mounted in beehives. Current Trends in Natural Sciences, 7, 13, University of Pitesti, 2018, ISSN:2284-953X, 47-53.
9. **Dineva K.,** Atanasova T., Computer System Using Internet of Things for Monitoring of Bee Hives, SGEM GeoConference, 27 - 29 November, 2017, ISSN:1314-2704, DOI:10.5593/SGEM_GeoConference, 2017, Vol. 17, Issue 63, pp. 169-176, **Q4**, **SJR:0.232.**
10. **Dineva K.,** Atanasova T., Model of Modular IoT-based Bee-Keeping System. European Simulation and Modelling Conference ESM'2017, EUROSIS-ETI, 2017, ISBN:978-492859-00-6, 404-406, **Scopus.**
11. **Dineva K.,** Analytical review of existing computer systems for monitoring of beehives, Proc. International Scientific Conference UNITECH'2017, 17-18 November 2017, Gabrovo, Bulgaria, ISSN: 1313-230X, pp. II-148-II-152.
12. **Dineva K**., Internet of Things in Help of Sustainable Agricultural Development, International Conference AUTOMATICS AND INFORMATICS'2017, 4-6 October 2017, Sofia, Bulgaria, John Atanasoff Society of Automatics and Informatics, Sofia, Bulgaria, 2017, ISSN:1313-1850, pp.309-312.

Citations

I. Dineva, K., Atanasova, T.: Model of Modular IoT-based Bee-Keeping System, ESM'2017, Lisbon, EUROSIS-ETI, 404-406 (**2017**).

Citing works:

1. Todor Balabanov, Ivan I. Blagoev, Zornitsa Atanassova, Greedy Genetic Algorithm Hybrid Solution of 1D Stock Cutting Problem, International Scientific Conference UNITECH 2018, November **2018**, Gabrovo, Bulgaria, ISSN 1313-230X, pp.307-312.
2. S. Šakanović, N. Dogru, D. Kečo and J. Kevrić, "Short-Term Prediction of Honey Production in Bosnia and Herzegovina using IoT," *2019 8th Mediterranean Conf. on Embedded Computing (MECO)*, Budva, Montenegro, **2019**, pp. 1-4. doi: 10.1109/MECO.2019.8760012.
3. Johannes Pirhonen. "A data transmitter using the GSM network", Bachelor's thesis, University of Tampere, Finland, Bachelor's Degree Program in Information and Electrical Engineering, Electrical Engineering, December **2019.**
4. Riste Poposki, Dejan Gjorgjevikj, Precision Apiculture - IoT System for remote monitoring of honeybee colonies, 17th International Conference on Informatics and Information Technology - CIIT2020, Ss. Cyril and Methodius University in Skopje, **2020.**

II. Dineva, K., Atanasova, T.: Computer System Using Internet of Things for Monitoring of Beehives, Vienna, SGEM GeoConference, vol. 17, Issue 63, 169-176 (**2017).**

Citing works:

5. Balabanov T., I. I. Blagoev, Z. Atanassova, Greedy Genetic Algorithm Hybrid Solution of 1D Stock Cutting Problem, International Scientific Conference UNITECH 2018, November **2018,** Gabrovo, Bulgaria, ISSN 1313-230X, pp.307-312.
6. Braga, A.R., Rabelo, J.C., Callado, A.C., da Rocha, A.R., Freitas, B.M., Gomes, D.G. Beenotified! a notification system of physical quantities for beehives remote monitoring | [Beenotified! um sistema de notificac¸˜oes de grandezas f´ısicas para monitoramento remoto de Colmeias de abelhas], *Revista de Informatica Teorica e Aplicada*, 27(3), pp. 50-61, **2020**

III. Dineva, K., Internet of Things in Help of Sustainable Agricultural Development, International Conference AUTOMATICS AND INFORMATICS'2017, 4-6 October 2017, Sofia, Bulgaria, John Atanasoff Society Of Automatics And Informatics, ISSN:1313-1850, pp.309-312, (**2017**).

Citing work:

7. Atanasova T., N. Bakanova, I. Blagoev, Analysis of Data from OIS To Discover and Model Process-Oriented Information, NVU "Vasil Levski", 14-15 June **2018**, Tom 9, pp. 106-111.

IV. Dineva, K., Atanasova, T., OSEMN Process For Working Over Data Acquired By Iot Devices Mounted In Beehives, Current Trends in Natural Sciences, 7, 13, University of Pitesti, **2018**, ISSN:2284-953X, 47-53, http://www.natsci.upit.ro.

Citing works:

8. Jae Deok Son, Sooho Lim, Dong-In Kim, Giyoun Han, Rustem Ilyasov, Ural Yunusbaev and Hyung Wook Kwon, Automatic Bee-Counting System with Dual Infrared Sensor based on ICT, *Journal of Apiculture* 34(1): 47-55 (**2019**) DOI: 10.17519/apiculture.2019.04.34.1.47
9. Blagoev I., Application of Time Series Techniques for Random Number Generator Analysis, Proceedings of XXII Int. Conference DCCN 2019, September 23-27, **2019**, Moscow, Russia, pp. 437-446. ISBN 978-5-209-09683-2, 2019.
10. Yadhunath R. , S. Srikanth, A. Sudheer and S. Palaniswamy, "Identification of Criminal Activity Hotspots using Machine Learning to Aid in Effective Utilization of Police Patrolling in Cities with High Crime Rates, " 2019 4th International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS), IEEE, Bengaluru, India, **2019**, pp. 1-6.

11. Braga, A. R., Gomes, D. G., Rogers, R., Hassler, E. E., Freitas, B. M., Cazier, J. A. "A method for mining combined data from in-hive sensors, weather and apiary inspections to forecast the health status of honeybee colonies", *Computers and Electronics in Agriculture*, Volume 169, February **2020.**

12. Braga A.R., Gomesa D. G., Freitas B.M., Cazier J.A. "A cluster-classification method for accurate mining of seasonal honey bee patterns", *Ecological Informatics*, Elsevier, **2020**

13. Braga, Antonio Rafael, Modelos de Classificação Para Predição do bem Estar de Colônias da Abelha Apis Mellifera, Doutorado em Engenharia de Teleinformática, Fortaleza **2020**

14. Da Silva, Daniel; Rodrigues, Ícaro; Braga, Antonio; Nobre, Juvêncio; Freitas, Breno; Gomes, Danielo. An Autonomic, Adaptive and High-Precision Statistical Model to Determine Bee Colonies Well-Being Scenarios. In: WORKSHOP DE COMPUTAÇÃO APLICADA À GESTÃO DO MEIO AMBIENTE E RECURSOS NATURAIS (WCAMA), 11, 2020, Evento Online. Anais do XI Workshop de Computação Aplicada à Gestão do Meio Ambiente e Recursos Naturais. Porto Alegre: Sociedade Brasileira de Computação, june **2020**, p. 31-40. ISSN 2595-6124. DOI: https://doi.org/10.5753/wcama.2020.11017.

V. Dineva, K., Atanasova, T. Applying machine learning against beehives dataset. SGEM 2018, 18, 6.2, SGEM 2018, (**2018)**, ISBN:978-619-7408-51-5, ISSN:1314-2704, DOI:10.5593/sgem2018/6.2, 35-42.

Citing works:

15. Braga, A. R., Gomes, D. G., Rogers, R., Hassler, E. E., Freitas, B. M., Cazier, J. A. "A method for mining combined data from in-hive sensors, weather and apiary inspections to forecast the health status of honeybee colonies", *Computers and Electronics in Agriculture*, Volume 169, February **2020.**

16. Braga, Antonio Rafael, Modelos de Classificação Para Predição do bem Estar de Colônias da Abelha Apis Mellifera, Doutorado em Engenharia de Teleinformática, Fortaleza **2020**

VI. Dineva K., Atanasova T., Security in IoT Systems, SGEM 2019, Vol. 19, Informatics, Geoinformatics and Remote Sensing, Issue 2.1, ISSN 1314-2704, pp. 576-577, (**2019)**

Citing work:

17. P. Panev, S. Dimitrov, Innovative Technology for Increasing the Efficiency in Tubular Furniture Production Machine, 8th International Conference on Advanced Technologies (ICAT'19), August 26-30, 2019, Sarajevo, Bosnia and Herzegovina, E-ISBN: 978-605-68537-4-6 pp. 338-341**, 2019.**

VII. Dineva, K., Atanasova, T. Regression analysis on data received from modular IOT system. ESM'2019, EUROSIS-ETI, (**2019)**, ISBN: 978-9492859-09-9, pp. 114-120.

Citing work:

18. Tomov, P., Zankinski, I., Balabanov, T. "Training of Artificial Neural Networks for Financial Time Series Forecasting in Android Service and Widgets", Scientific journal "*Problems of Engineering Cybernetics and Robotics*", vol. 71, pp. 50-56**, 2019**.

VIII. Dineva, K., Atanasova, T. ICT-based Beekeeping using IoT and Machine Learning. Distributed Computer and Communication Networks, 21-st International Conference, DCCN 2018, 919, Springer, (**2018)**, ISBN:978-3-319-99446-8, ISSN:1865-0929, https://doi.org/10.1007/978-3-319-99447-5, pp. 132-143.

Citing work:

19. Pešović, U., D. Marković, S.Ď. Đurašević. "Remote monitoring of beehive activity*". Acta Agriculturae Serbica*, Vol. XXIV, 48 (**2019**); 157-165

Bibliography

1. **Cohn J.** Scrum Mastery + Agile Leadership: The Essential and Definitive Guide to Scrum and Agile Project Management [Book]: Jeff Cohn, 2019.
2. **Manel D. Fraser S.** Agile Processes in Software Engineering and Extreme Programming – Workshops [Conference]: 20th International Conference on Agile Software Development. - Montréal : Springer, 2019.
3. **R. Hertzog R. Mas** The Debian administrator's handbook: Debian Jessie from Discovery to Mastery [Book]: Freexian, 2015.
4. **Alekseev I. Nikitinskiy M.** Eventbus module for distributed openflow controllers [Conference]: 17th Conference of Open Innovations Association (FRUCT), IEEE, 2015.
5. **Hesse G. Matthies Ch., Rabl T., Uflacker M.** How Fast Can We Insert? A Performance Study of Apache Kafka [Journal]: Cornell University, 2020.
6. **Japikse P. Grossnicklaus K., Dewey B.** Building the Spy Store Web Application with Angular [Journal]: Berkeley, CA : Apress, 2020.
7. **Brada A. Gomes D., Rogers R., Hassler E., Freits B., Cazier J.** A method for mining combined data from in-hive sensors, weather and apiary inspections to forecast the health status of honey bee colonies [Journal]: Elsevier, 2020. - Vol. 169.
8. **Beecham Research** Smart Farming: The sustainable way to food [Report]: Beecham Research, 2017.
9. **Kotler Ph.** Philip Kotler's Contributions to Marketing Theory and Price [Journal]: Emerald Group Publishing limited. - 2011. - pp. 87-120.
10. **Jazdi N.** Cyber physical systems in the context of Industry 4.0 [Journal]. - Romania : IEEE International Conference on Automation, Quality and Testing, Robotics, 2014.
11. **Atanasova T.** Methods for Processing of Heterogeneous Data in IoT Based Systems [Conference]: DCCN. - Moscow : Springer, 2019. - pp. 524-535.
12. **Sorrell S.** The Internet Of Things: Consumer, Industrial & Public Service 2018-2023 [Report]: Juniper Research, 2018.
13. **Atwal H.** DataOps Technology. In: Practical DataOps [Book]: Apress, 2020.
14. **Anoshin D. Shirokov D., Strok D.** Getting Started with Cloud Analytics, Jumpstart Snowflake [Book]: Apress, 2020.
15. **Singh J. Saravanan B., Prased V.** [Patent]: 16/147, 976. - US, 2019.
16. **Guhr S. Martenson J., Laser H.** Data Science as a Service - Prototyping for an Enterprise Self-Service Platform for Reproducible Research [Conference]: IARIA - The Fifth International Conference on Fundamentals and Advances in Software Systems Integration: FASSI, 2019.
17. **Howell K.** An introduction to the philosophy of methodology [Book] : SAGE, 2013.
18. **Saif S. Garba A., Awwalu J., Arshad H., Zakaria L.** Performance Comparison of Min-Max Normalisation on Frontal Face Detection Using Haar Classifiers [Journal]: PERTANIKA, 2017.
19. **Samariya D. Aryal S., Ting K.** A new effective and efficient measure for outlying aspect mining [Journal]: Cornell University, 2020.
20. **Waltenburg E. McLauchlan W.** An Introduction to Exploratory Data Analysis [Book Section]: Exploratory Data Analysis: A primer for undergraduates. Purdue e-Pubs, 2012.
21. **Kolchakov K. Monov V.** Query Conflicts In An Algoorithm For NoN Conflict Scheduling For Crossbar Commutator [Conference] : International Conference Automatics and Informatics`2018. - Sofia : SAI, 2018.

22. **Mease D. Wyner A. J. and Buja A.** Boosted Classification Trees and Class Probability Estimation [Journal]: JMLR, 2007.
23. **Agarwal K. Uniyal P., Virendrasingh S., Duff V.** Spam Mail Classification using Ensemble and Non- ensemble Machine Learning Algorithms [Book Section]: Springer SIST, Springer, 2020.
24. **Tewari S.** Ensemble Methods: An intelligent Modeling Approach [Conference]: EAGE Annual 2020, EAGE, 2020.
25. **Balabanov T. Ivanov S., Ketipov R.,** Solving Combinatorial Puzzles with Parallel Evolutionary Algorithms [Conference]: Large-Scale Scientific Computing: 12th International Conference, LSSC 2019, Springer, 2020.
26. **Vanwinkelen G. Blockeel H.** On estimating Model Accuracy with repeated Cross-Validation [Conference]: Belgian-Dutch Conference on Machine Learning, 2012.
27. **Balabanov T. Atanasova T., Blagoev I.,** Activation Function Permutation for Multilayer Perceptron Training [Conference]: International Conference on Big Data, Knowledge and Control Systems Engineering, Sofia : IEEE, 2018.
28. **Abraham T. Parasher Sh.** Hands-on Machine Learning with Azure [Book]: Packt, 2018.
29. **Azure** Microsoft Azure Machine Learning Studio [Online], 2020. https://studio.azureml.net/.
30. **Braga A. Gomes D., Rogers R., Hassler E., Freits B., Cazier J.,** A method for mining combined data from in-hive sensors, weather and apiary iinspections to forecast the health status of honey bee colonies [Journal]: Elsevier, 2020. - Vol. 169.

Design: GOST standard