INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2026, VOL. 72, NO. 2, PP. 1-6
Manuscript received April 14, 2026; revised April 2026. doi: 10.24425/ijet.2026.157939

The Internet of Things as a semantic infrastructure – integration of sensor data with distributed knowledge systems in Edge – cloud environments

Mateusz Cieślak

Abstract—**The rapid growth of the Internet of Things (IoT) generates large volumes of heterogeneous and context-dependent data from distributed sensing devices, posing challenges for integration and analysis. This paper proposes a conceptual model of IoT as a semantic infrastructure, integrating sensor data with distributed knowledge systems in edge–cloud environments. The approach utilizes semantic technologies, including RDF, OWL, and SSN/SOSA ontologies, to enable unified data representation and contextual enrichment. The proposed model improves interoperability, supports efficient distributed processing, reduces latency, and enhances scalability, enabling the development of intelligent and adaptive IoT systems.**

***Keywords*—Internet of Things (IoT); semantic IoT; RDF/OWL; SSN/SOSA; data integration; edge–cloud computing; artificial intelligence; interoperability**

I. Introduction

THE development of the IoT constitutes one of the key research directions in the field of distributed systems and data processing. The dynamic growth in the number of IoT devices, along with the increasing complexity of communication environments, renders traditional data processing approaches insufficient, particularly in terms of scalability, integration, and automation of decision-making processes. A significant challenge remains the lack of unambiguous data interpretation and the limited ability to effectively utilize the available data. The literature indicates that the use of semantic technologies, such as RDF, OWL, and SSN/SOSA ontologies, enables the assignment of contextual meaning to data and facilitates their integration with distributed knowledge systems [1], [3]. This approach supports the interoperability of IoT systems and their integration with artificial intelligence methods; however, it also introduces challenges related to scalability and real-time processing [2].

In parallel, edge–cloud architectures are being developed to enable distributed data processing between the edge layer and cloud computing environments, contributing to reduced latency and improved system efficiency. The integration of semantic technologies with this approach is identified as a significant direction for the development of modern IoT systems [4]. However, existing solutions primarily focus either on semantic data representation or on optimizing processing in edge–cloud architectures, rarely combining these aspects in a coherent manner.

Despite technological advancements, significant limitations persist, including the lack of cohesive integration of semanticmodeling with distributed processing, limited scalability, and challenges in achieving efficient real-time reasoning. This indicates the existence of a research gap related to the absence of an integrated approach to IoT data processing that combines semantic data modeling with edge–cloud architecture.

This paper proposes a conceptual data integration model that addresses these limitations and enables efficient data processing in distributed environments.

II. Characteristics of Sensor Data in IOT Systems

Sensor data generated in IoT systems exhibit specific properties that significantly influence their processing, integration, and analysis. Unlike traditional data, they are continuously generated by distributed sensing devices and are characterized by high variability and strong dependence on environmental context.

One of the key characteristics of IoT data is their heterogeneity, resulting from the diversity of sensor types, communication protocols, and data formats. These data may take the form of numerical values, time-series streams, or event-based records, which complicates their standardization and integration. Another important property is the large volume and high velocity of data generation. IoT systems produce data in real time, which requires the use of solutions enabling efficient and low-latency processing and analysis. In the context of edge–cloud architecture, it becomes essential to distribute processing tasks between edge and cloud layers.

Sensor data are also often incomplete and uncertain due to measurement errors, transmission disturbances, or hardware limitations. Missing or imprecise data may reduce the quality of analysis and lead to incorrect decisions in IoT systems. An additional important aspect is data contextuality, which includes information such as the time of measurement, location, and environmental conditions. Without contextual information, sensor data lose their interpretability, limiting their applicability in analytical and decision-making systems.

Given these characteristics, traditional data processing approaches prove insufficient. Therefore, it is necessary to apply methods that enable unified data representation and semantic enrichment, forming the basis for further integration with distributed knowledge systems.

III. Edge–Cloud Architecture in Distributed Internet of Things Systems

Edge–cloud architecture [5], discussed below, represents one of the key approaches to data processing in IoT systems, enabling

Author is with the Warsaw School of Computer Science, Lewartowskiego 17, 00-169 Warsaw, Poland (e-mail: maticieslak7@gmail.com)

the distribution of computational processes between the edge layer and the cloud environment.

The architecture consists of three main layers: the device layer, the edge layer, and the cloud layer. The device layer includes various data sources, such as mobile devices, autonomous systems, and industrial infrastructure, which generate sensor data. These data are transmitted to the edge layer, represented by edge nodes or local servers, where preliminary processing is performed. The cloud layer, acting as a central computational environment, is responsible for further data processing and integration, while also enabling bidirectional communication with the edge layer.

The edge layer is responsible for processing data close to their source, i.e., directly on IoT devices or in their proximity. At this level, operations such as data filtering, aggregation, normalization, and preliminary analysis are performed, which reduces the volume of data transmitted to the cloud and minimizes latency. This approach is particularly important in systems requiring rapid responses to changing environmental conditions. Additionally, local processing at the edge improves system reliability, as it allows basic operations to be performed even in the case of limited network connectivity.

The cloud layer provides high scalability and access to substantial computational resources, enabling more complex operations such as large-scale data analytics, machine learning model training, and integration with distributed knowledge systems. This layer also supports global reasoning, long-term data storage, and historical data analysis. Furthermore, cloud environments enable centralized system management and facilitate the integration of data from multiple distributed sources.

The integration of edge and cloud layers allows for optimal distribution of processing tasks, leading to improved efficiency of IoT systems. Preliminary processing at the edge reduces latency and communication overhead, while the cloud layer enables advanced analytical and decision-making processes. This hybrid approach makes it possible to combine fast local responses with complex global analysis.

Another important advantage of edge–cloud architecture is its scalability and flexibility. As the number of IoT devices increases, the system can dynamically distribute computational load between edge and cloud components. This ensures stable performance even under high data throughput and supports the development of large-scale IoT systems.

Despite its advantages, the edge–cloud architecture does not address the issue of unambiguous data interpretation or semantic integration. Data processed within this architecture may remain heterogeneous and lack contextual meaning, which limits their usability in knowledge-based systems. Moreover, without a semantic layer, it is difficult to ensure consistent interpretation of data across different system components. Therefore, it is necessary to complement the edge–cloud approach with semantic data modeling mechanisms, which form the basis for further considerations.

IV. SEMANTIC APPROACH IN IOT SYSTEMS – DATA INTEGRATION USING ONTOLOGIES AND SEMANTIC MAPPING

In IoT systems, one of the key challenges is ensuring [6] clear data interpretation and effective integration across heterogeneous sources. Data generated by IoT devices are highly diverse in terms of format, structure, and context, which limits their direct use in analytical and decision-making systems. Traditional data processing approaches, based on syntactic representation, do not capture the meaning of information or the relationships between data elements, resulting in reduced interoperability.

To address these limitations, semantic technologies are increasingly applied to support formal data representation and integration with knowledge models. Heterogeneous data [7] originating from physical IoT devices are transformed into structured forms and enriched with metadata describing their context, origin, and semantics. This transformation is typically performed using standards such as RDF, which enables data representation as triples, and OWL, which supports the formal modeling of domain knowledge.

Domain-specific ontologies play a crucial role by defining concepts, attributes, and relationships within a given domain. However, since different systems may rely on distinct ontologies, semantic integration requires the use of ontology alignment and mapping techniques. These methods establish relationships between concepts across different ontologies using schema matching, rule-based, and similarity-based approaches, enabling consistent interpretation of data from heterogeneous sources.

A central element of this approach is the concept of semantic similarity, which supports the comparison and integration of data across different domains. Semantic similarity measures evaluate the degree of relatedness between concepts based on their position within an ontology, shared properties, or contextual usage. This allows the identification of equivalent or related entities, even when they are described using different terms or structures.

As a result, raw IoT data are transformed into semantically enriched representations that can be integrated within distributed knowledge systems. This enables advanced data processing capabilities, including reasoning, inference, and context-aware analysis. Furthermore, this approach highlights the interaction between the physical layer (devices and sensors) and the virtual layer (data models and ontologies), where semantic technologies act as a bridge ensuring consistent data interpretation and system compatibility.

The semantic approach assumes that data are not treated merely as numerical or textual values, but as elements of a broader semantic structure in which relationships and context play a key role. The fundamental technologies supporting this approach include the Resource Description Framework (RDF) and the Web Ontology Language (OWL). RDF [8] enables data representation in the form of semantic triples (subject–predicate–object), allowing the modeling of relationships and the construction of knowledge graphs. OWL [9] extends these capabilities by introducing more expressive modeling constructs and reasoning mechanisms, enabling the automatic inference of new knowledge based on existing data.

An important component of semantic technologies is domain ontologies, which provide a standardized description of concepts and relationships within a given domain. In the context of IoT systems, ontologies such as SSN/SOSA [10] play a crucial role by enabling formal descriptions of sensors, observations, and physical properties. Their use ensures unified data representation and supports compatibility across different systems and platforms.

By applying semantic technologies, it becomes possible to assign meaning to data, combine information from multiple sources, and perform automated reasoning. As a result, raw sensor data can be transformed into structured knowledge that can be used in advanced analytical and decision-support systems.

Despite its advantages, the semantic approach introduces challenges, including increased computational complexity and difficulties related to real-time processing. In particular, reasoning operations can be resource-intensive, which limits their applicability in the edge layer of IoT systems.

Therefore, it is necessary to adopt an architecture that enables efficient distribution of processing tasks. The integration of the semantic approach with edge–cloud architecture allows preliminary data processing to be carried out at the edge, while more complex semantic operations are executed in the cloud. This approach forms the foundation of the proposed IoT data processing model.

V. Identification and Detailed Analysis of Limitations of Existing Solutions

Despite the rapid development of IoT technologies and semantic data processing methods, existing solutions still face significant limitations that affect their efficiency, scalability, and applicability in distributed environments. As summarized in Table I, these limitations highlight key challenges in current IoT data processing approaches.

TABLE I
IDENTIFICATION AND ANALYSIS OF THE LIMITATIONS OF EXISTING SOLUTIONS IN THE CONTEXT OF SEMANTIC IOT DATA PROCESSING

Area	Limitations of Existing Solutions	Implications for IoT and Knowledge-Based Systems
Complexity of Semantic Data Processing	High computational overhead of RDF/OWL and complex ontology models	Performance degradation and limitations in real-time processing
Scalability	Challenges in processing large volumes of data	Limitations in large-scale device environments
Processing time	Latency resulting from semantic reasoning	Limited real-time responsiveness
Cloud-based architecture	Centralization of data processing	High latency
Data heterogeneity	Diversity of data formats and models	Data integration challenges
Lack of ontologies	Inconsistent semantic models	Difficulties in data interpretation
Integration with artificial intelligence (AI)	Lack of unified models integrating AI and semantic technologies	Underutilized analytical potential
Data fragmentation	Isolated data silos	Limited analytical capabilities
Edge computing	Limited utilization of edge computing	Overloading of the cloud layer

One of the main challenges is the high complexity and computational overhead associated with the use of semantic technologies such as RDF and OWL. Processing large-scale graph-based data requires substantial computational resources, which limits their applicability in IoT systems generating data continuously and at scale. Moreover, semantic models are often overly complex, which restricts their use in real-time applications.

Another important limitation concerns scalability. As the number of IoT devices and the volume of generated data increase, the complexity of their integration and processing also grows. Ensuring semantic interoperability in heterogeneous IoT environments remains a significant challenge, particularly when integrating data from multiple sources and systems.

A further challenge is the limited capability of systems to process data in near real time. Although semantic reasoning mechanisms provide high expressiveness, they introduce latency that may be unacceptable in applications requiring immediate response. This issue is particularly evident in cloud-centric architectures, which do not always meet low-latency requirements.

Data heterogeneity and the lack of standardized domain ontologies hinder data integration and automated processing. The diversity of data formats, models, and communication interfaces leads to the creation of isolated data silos, limiting the effective use of information in analytical systems.

IoT resources highlight the limited integration of semantic technologies with artificial intelligence methods. Despite the potential benefits of combining these approaches, there is still a lack of unified models enabling their effective cooperation in IoT environments.

The identified limitations indicate the existence of a research gap related to the lack of an integrated IoT data processing model that combines the semantic approach with edge–cloud architecture, while ensuring high interoperability, scalability, and efficient near real-time processing.

VI. Concept of Research on Semantic IoT Data Processing in Edge–Cloud Architecture

The proposed research concept defines a formal, integrated model for IoT data processing that combines semantic data representation with a distributed edge–cloud computing architecture. The model provides a unified framework for handling heterogeneous data sources while ensuring interoperability, scalability, and computational efficiency. From a formal perspective, the model can be interpreted as a multi-layer data processing pipeline, defined as an ordered sequence of transformation functions:

$$F = \{f_1, f_2, \dots, f_n\}$$

where each function maps input data from one representation space to another, progressively increasing the level of abstraction. In this context, each layer of the architecture is responsible for a well-defined subset of transformations, ensuring a clear separation of concerns and enabling modular optimization.

The architecture consists of four complementary layers: the physical layer, the edge layer, the semantic processing layer, and the cloud layer. This layered structure reflects the natural lifecycle of IoT data, from acquisition and preprocessing to structured representation and analysis. At the same time, the modular decomposition of the system enables independent scaling, replacement, and optimization of individual components, which is particularly important in large-scale, distributed IoT deployments.

The physical layer is responsible for deterministic data acquisition from heterogeneous IoT devices such as environmental sensors, embedded systems, and measurement units. In the proposed model, each device generates data according to a predefined schema that includes at least the sensor identifier, timestamp, measurement value, and measurement type. This schema can be interpreted as a minimal canonical representation of an observation, ensuring syntactic consistency and reducing integration complexity in subsequent processing stages. Data transmission is performed using the lightweight JSON format over the MQTT protocol, which minimizes communication overhead and supports reliable operation in constrained environments characterized by limited bandwidth, energy, and computational resources.

The edge layer performs a critical role in reducing data volume and preparing it for further processing by implementing a formally defined set of preprocessing operations. These operations can be described as a composite transformation function:

$$f_{edge}$$

consisting of validation, normalization, aggregation, and contextualization. Validation removes incomplete, inconsistent, or corrupted records, thereby ensuring data quality. Normalization unifies measurement units and data types, enabling consistent interpretation and comparability across heterogeneous sources. Aggregation reduces data granularity using time-based or event-based windows while preserving essential statistical characteristics such as mean, variance, or extrema. Contextualization enriches data with additional metadata, such as device location, operational state, or measurement category.

An additional mechanism introduced in the model is lightweight semantic annotation at the edge level, which can be formalized as a partial mapping function:

$$f_{sem}^{edge}$$

This function assigns selected attributes of the data to predefined categories using a constrained vocabulary. The objective of this operation is to preserve essential contextual information while maintaining low computational complexity.

The semantic processing layer constitutes the core component of the model and is responsible for transforming structured data into a graph-based representation. This process is defined as a mapping function:

$$f_{rdf}: D_{JSON} \rightarrow G_{RDF}$$

which converts normalized JSON records into sets of RDF triples. The mapping establishes explicit correspondences between input data fields and ontology elements, particularly classes and properties defined in the SSN/SOSA standard. Each observation is represented as a subgraph describing relationships between the sensor, the observed property, the measurement value, and the timestamp.

The model assumes incremental construction of the knowledge graph through the generation and integration of partial graphs G_i, such that the global graph is defined as:

$$G = \bigcup_i G_i$$

This approach supports scalable processing and continuous updates of streaming data.

The cloud layer is responsible for executing computationally intensive operations and ensuring system-wide scalability. It provides persistent storage of RDF graphs in triplestore repositories, supports SPARQL query execution, and enables both batch and real-time stream analytics. A clear distinction is made between graph-based processing and numerical data analysis. Graph-based operations are used for structured querying and rule-based processing, while numerical representations derived from the same data are used for statistical analysis.

The model incorporates a bidirectional feedback mechanism between the cloud and edge layers, which can be formalized as a control function:

$$f_{feedback}: A \rightarrow P$$

where A denotes analytical results and P represents configuration parameters for edge processing. These parameters include aggregation intervals, filtering thresholds, and preprocessing rules. The feedback is transmitted via MQTT, enabling controlled system adaptation without modifying the system structure.

From an implementation perspective, the model is based on a technological stack including ESP32-based edge nodes, MQTT communication infrastructure, RDF processing libraries, ontology management tools, scalable cloud platforms, graph-based storage systems, and rule engines. The selection of these technologies is aligned with the defined processing pipeline.

The evaluation of the proposed model will be conducted using real-world sensor data streams within a controlled experimental environment. The evaluation framework includes measurable metrics such as processing latency, data reduction efficiency, correctness of transformations, scalability of graph processing, and system behavior under limited connectivity conditions.

The contribution of this work lies in the explicit definition of a complete and structured IoT data processing pipeline, in which individual stages of data handling are formally described in terms of transformation functions, data representations, and clearly assigned layer responsibilities. The proposed model introduces a consistent framework that integrates data acquisition, preprocessing, transformation, and analysis within a unified architectural structure.

A key aspect of the contribution is the formalization of data transformations as an ordered sequence of functions, which enables precise specification of data flow and dependencies between processing stages.
This approach allows for clear identification of the role of each layer, including the physical layer responsible for data generation, the edge layer handling preprocessing operations, the transformation layer responsible for structured representation, and the cloud layer performing advanced processing tasks.

The model also contributes by introducing a systematic approach to handling heterogeneous data sources. By defining a consistent representation of input data and structured transformation mechanisms, it reduces integration complexity and supports interoperability across system components. Furthermore, the explicit separation of processing stages enables modular system design, allowing individual components to be independently developed, optimized, or replaced without affecting the overall system structure.

Another important element of the contribution is the inclusion of a feedback mechanism that enables dynamic adjustment of processing parameters. This allows the system to adapt to changing conditions while preserving the integrity of the processing pipeline. The formal definition of this mechanism supports controlled system behavior and facilitates performance tuning.

In addition, the proposed model provides a basis for systematic evaluation. By defining clear transformation stages and measurable outputs, it enables the assessment of system performance using well-defined metrics such as latency, data reduction efficiency, and scalability. This makes the model suitable not only for conceptual analysis but also for practical implementation and experimental validation.

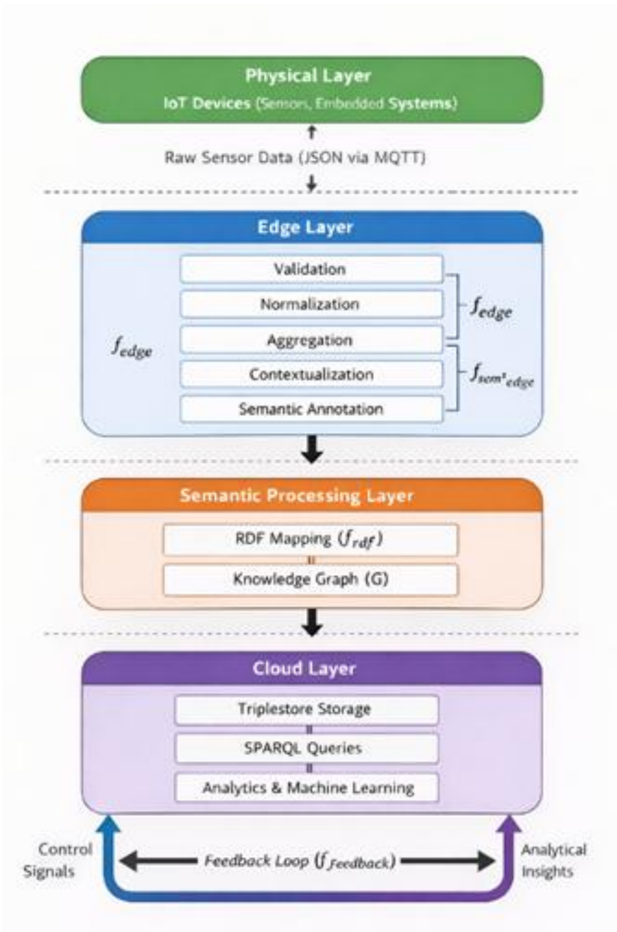

Fig.1. Flow diagram of the proposed IoT data processing model.

The flow diagram defines a structured data processing pipeline in an IoT environment, integrating edge computing, semantic data processing, and cloud-based analytics. Raw sensor data, transmitted in lightweight formats such as JSON over MQTT, is inherently heterogeneous and requires further transformation.

At the edge level, data is processed by a transformation function f_{edge}, which consists of operations such as validation, normalization, aggregation, contextualization, and semantic annotation. These steps improve data quality, reduce redundancy, and enrich the data with contextual and semantic information.

Subsequently, the data is mapped into a formal semantic representation through a function f_{rdf} and organized into a knowledge graph G, enabling interoperability and machine interpretability.

In the final stage, data is stored in a triplestore and processed using SPARQL queries as well as analytics and machine learning methods, enabling the extraction of high-level insights.

A feedback function $f_{feedback}$ propagates analytical results back to earlier stages in the form of control signals, supporting adaptive and optimized system behavior.

VII. Opportunities for the Use of Artificial Intelligence in the Proposed Approach

In the context of the proposed approach, which considers the IoT as a semantic infrastructure, the application of artificial intelligence (AI) methods plays a crucial role in the processing of sensor data. The integration of machine learning techniques with semantic technologies enables not only data analysis but also their interpretation within a domain-specific context, forming the foundation for the development of systems capable of autonomous decision-making.

Data generated by IoT devices are typically stream-oriented, often incomplete, noisy, and highly dependent on environmental context. Therefore, the application of AI enables their effective processing through the identification of hidden patterns, error reduction, and completion of missing information. In particular, the use of generative models and data imputation methods allows for the reconstruction of missing values, significantly improving data quality for subsequent analytical stages.

An important area of AI application is anomaly detection, which enables the identification of unusual events and deviations from normal behavior patterns. In IoT systems, such mechanisms can be used for environmental monitoring, fault detection, and early warning of potential risks. The use of unsupervised and semi-supervised learning methods allows anomaly detection even in the absence of complete training datasets.

Another significant direction is event prediction, which enables forecasting future system states based on historical data and current observations. Predictive models, including neural networks and sequence-based models, can be applied to forecast environmental parameter changes, optimize resource usage, and support decision-making processes. In IoT systems, this enables a transition from reactive to predictive operational models.

An important aspect is the integration of AI with the semantic approach. Combining machine learning models with ontologies and knowledge graph representations enables the development of hybrid systems that integrate statistical analysis capabilities with semantic interpretability.

This approach enhances data understanding and increases the transparency of decision-making processes.

Within the edge–cloud architecture, AI can be applied at multiple levels of data processing. At the edge layer, lightweight models can be deployed to enable rapid decision-making and reduce latency. In contrast, the cloud layer supports more complex operations, such as model training, large-scale data analysis, and integration with knowledge systems. This distribution of functions allows for efficient utilization of computational resources and improves system scalability.

Despite its advantages, the application of AI in IoT systems also presents challenges, including limited computational resources at the edge, the need to ensure low latency, and the requirement for model interpretability. In particular, in critical systems, ensuring transparency and reliability of decisions is essential, which necessitates the integration of AI methods with formal knowledge models.

The integration of artificial intelligence with semantic technologies and edge–cloud architecture constitutes a key element in the development of modern IoT systems. This approach not only improves the quality of processed data but also enables the creation of adaptive systems capable of contextual data interpretation and supporting advanced analytical and decision-making processes.

VIII. Conclusion and Future Work

The progress of research in the field of the IoT and semantic technologies indicates that the integration of sensor data with distributed systems constitutes an important direction in the evolution of modern data processing architectures. The analysis presented in this paper shows that extending traditional data processing approaches with additional representation layers leads to changes in how data are structured, integrated, and processed in distributed environments.

The main result of this work is the development of a formal model for IoT data processing, defined as a multi-layer transformation pipeline. The model is based on an ordered set of transformation functions, where each layer is responsible for a specific stage of data processing and increases the level of abstraction. This approach enables a clear definition of data flow, separation of responsibilities, and independent optimization of system components.

The proposed architecture consists of four layers: the physical layer, the edge layer, the data transformation layer, and the cloud layer. The physical layer is responsible for data acquisition in a unified format, the edge layer performs initial preprocessing operations such as validation, normalization, and aggregation, the transformation layer maps data into a structured representation, and the cloud layer handles computationally intensive processing tasks. This structure allows efficient distribution of processing load and reduction of latency.

Compared to approaches based solely on syntactic data processing, the proposed model integrates data processing with an edge–cloud architecture, enabling more efficient resource utilization and supporting both time-critical operations and large-scale analytics. The distribution of processing tasks across layers allows flexible adaptation to different data characteristics and system requirements.

The model assumes a direct relationship between data originating from physical devices and their structured representation, enabling consistent processing across system components. As a result, data are treated not only as input values but as elements of a structured processing pipeline.

An important aspect of the model is the formal definition of processing functions, including preprocessing at the edge layer and transformation into a graph-based representation. In addition, a feedback mechanism between the cloud and edge layers is introduced, allowing dynamic adjustment of processing parameters without modifying the system structure.

Despite its advantages, the implementation of the model involves several challenges, including ensuring scalability in environments with a large number of IoT devices, reducing processing overhead, and integrating heterogeneous data sources. Another important issue is achieving efficient near real-time processing under distributed conditions.

Future research directions include the optimization of edge-level processing, performance evaluation under different load scenarios, and the development of methods for improving data quality, including handling incomplete data. Further work should also focus on the development of simulation and experimental environments for evaluating system behavior under realistic conditions.

Additional studies should consider the relationships between system layers and the impact of transformation functions on the overall processing outcome. In particular, coordination between local processing, cloud-based computation, and data transformation stages remains an important area of analysis.

In conclusion, the proposed model provides a structured and consistent approach to IoT data processing in distributed environments. The combination of layered processing and formally defined transformations enables the development of scalable, flexible, and practically applicable systems. Furthermore, the integration of semantic technologies enhances interoperability and supports intelligent, data-driven decision-making in complex IoT environments.

References

[1] M. Compton, „The SSN ontology of the W3C semantic sensor network incubator group," pp. 25-32, 2012.

[2] W. Shi, „Edge computing: Vision and challenges," *IEEE Internet of Things Journal,* 2016.

[3] P. Barnaghi, „Challenges for quality of data in smart cities.," *Journal of Data and Information Quality.,* 2015.

[4] M. Satyanarayanan, „The Emergence of Edge Computing," *IEEE,* 2017

[5] G. Jayavardhana, „Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems,* 2013.

[6] [Online]. Available: https://www.geeksforgeeks.org/system-design/edge-cloud-architecture-in-distributed-system/

[7] [Online]. Available: https://www.researchgate.net/figure/Semantic-interoperability-for-Internet-connected-objects_fig83_260712642.

[8] S. Staab, „An Extensible Approach for Modeling Ontologies in RDF(S)".

[9] M. Ruta, „A Format and Protocol for Machine-to-Machine OWL Data Streams," *IEEE,* 2025.

[10] K. Janowicz, „SOSA: A Lightweight Ontology for Sensors, Observations, Samples, and Actuators," *SSRN Electronic Journal,* 2018.