# Hardware Support in an Intelligent IT Diagnostic System for Alzheimer's Disease

Uladzimir Vishniakou
Belarusian State University of
Informatics and Radioelectronics
Minsk, Belarus
vish@bsuir.by

Yu Chuyue
Belarusian State University of
Informatics and Radioelectronics
Minsk, Belarus
1259017720@qq.com

***Abstract*—The report presents an integrated intelligent diagnostic framework combining semantically oriented knowledge representation based on ontologies, hardware acceleration using FPGA and OSTIS concepts in order to increase the efficiency of real-time processing for IT diagnostics of patients with Alzheimer's disease. An ontology has been developed for such patients, on the basis of which a knowledge base, a solver and an intelligent interface are presented. An FPGA accelerator is proposed, due to which an intelligent IT diagnostic system for patients with ASTHMA provides efficient collection, preprocessing, feature extraction and semantic analysis of voice data, which reduces diagnostic delays and increases decision-making efficiency by about an order of magnitude. The developed ontologies are consistent with the semantic requirements of the OSTIS knowledge management concepts. The developed hybrid CPU-FPGA architecture and the proposed user interface integrate knowledge base queries, hardware management, real-time monitoring and diagnostic conclusion, demonstrating improved user interaction, semantic transparency and operational efficiency.**

***Keywords*—ontology, Alzheimer's disease, semantic representation of knowledge, solver, knowledge base, intelligent interface, FPGA accelerator.**

## I. Introduction

At the present stage, Alzheimer's disease (AD), as the most common neurodegenerative disorder among the elderly population, has become a critical topic in the field of global public health [1]. An annual report released by the Alzheimer's Association, reveals the burden of Alzheimer's and dementia on individuals, caregivers, government and the nation's health care system [2], in 2024, Alzheimer's and other dementias will cost the nation 360 billion dollars, by 2050, these costs could rise to nearly 1 trillion dollars. Currently, diagnostic technologies for AD are undergoing a transformation from traditional clinical assessments to intelligent and digitalized frameworks.

Mainstream IT-based diagnostic approaches for AD currently integrate information technology with medical science. The primary methods and applications include deep learning-based medical imaging analysis, multimodal data fusion, digital cognitive assessment tools, voice and language analysis (NLP technologies) [3]. The development of an intelligent IT diagnostic system for AD, based on these mainstream methods, involves eight key stages: requirement analysis and planning, data acquisition and management [4], data preprocessing and feature engineering, among others. Based on the OSTIS platform, an IoT network architecture [5] is adopted to construct domain-specific semantic knowledge ontologies for AD, along with corresponding solvers and user interfaces, thereby enabling efficient voice data acquisition, processing, and semantic analysis to support early clinical diagnosis and intervention.

In the construction of an intelligent IT diagnostic system, hardware support is a fundamental requirement for data acquisition, real-time monitoring, and large-scale computation. Medical imaging (such as MRI and PET) cannot be achieved without hardware support from imaging devices in clinical settings [6], hardware accelerators such as FPGAs, equipped with programmable logic arrays, can execute convolutional operations, matrix multiplications, and other tasks in parallel, significantly reducing inference latency and power consumption [7].

OSTIS, as an open semantic technology framework designed for the development and implementation of intelligent systems, aims to reduce the complexity of interaction among heterogeneous computing systems by offering unified knowledge representation and a semantically compatible infrastructure, thereby facilitating knowledge sharing and collaborative reasoning. Given the real-time and high-efficiency requirements of data processing in medical diagnostic scenarios, the introduction of a hardware-software collaborative architecture provides a novel implementation pathway for OSTIS-based systems. This paper aims to present a framework that integrates hardware acceleration with intelligent information processing, incorporating medical ontologies and knowledge bases to support diagnostic decision-making for Alzheimer's disease.

## II. The Ontology

Ontology, as a tool, enables the formal and logical definition and classification of knowledge and concepts within a specific domain. By defining the attributes of domain-specific concepts and their interrelationships, ontology supports semantic linking and understanding between data, allowing computers to identify and reason about domain knowledge.

OSTIS (Open Semantic Technology for Intelligent Systems) is closely related to ontology [8]. It focuses on the design and operation of semantically compatible hybrid intelligent computing systems, employing ontological methods for structured knowledge representation and semantic standardization. It offers the technologies and tools necessary for building, managing, and utilizing knowledge graphs.

In intelligent systems for Alzheimer's Disease (AD) diagnostics, speech processing is one of the key sources for obtaining pathological clues from patients. Under the OSTIS technical framework of hardware-software co-design, ontology serves as a semantic knowledge representation tool. On the one hand, it provides standardized domain concepts and terminology to support medical term disambiguation, text feature extraction, and semantic relationship construction on the CPU side, ensuring consistency in data semantic representation. On the other hand, ontology clearly defines the relationships and reasoning rules among medical concepts, enabling the FPGA side to perform parallel reasoning and classification decisions based on structured knowledge. The semantic and data organization foundations provided by the following ontologies allow the system's hardware modules to work collaboratively with software and data in various stages of speech acquisition, preprocessing, feature extraction, and machine learning inference, while ensuring compliance and traceability within clinical settings.

***Patient Voice Characteristics Ontology***

:= [Define and describe the patient voice characteristics associated with the diagnosis of Alzheimer's Disease (AD).]

:= [Provides a structured description framework for sound data, ensuring that the collection, storage, and sharing of sound data all follow a set of clear standards.]

∋ *Characteristics of a sound sample*

⇒ *includes**:

{
- *Pitch*
  - := [The degree of highness or lowness of a sound, typically measured in Hertz (Hz).]
- *Volume*
- *Rhythm*
  - ⇒ *includes**:
    {
    - *Speech rate*
    - *Pause frequency*
    - *Overall voice dynamics*

    }
- *Pause Duration*
- *Speech Clarity*
- *Timbre*
- *Speech Coherence*
- *The change of sound*

}

∋ *Description of the Data Collection Environment*

⇒ *includes**:

{
- *Background Noise Level*
- *Recording Device Type*
- *Recording Format*
- *Recording Environment*
- *Recording Session Duration and Timing*
- *Recording Distance and Angle*
- *Speaker's Position and Posture*

}

∋ *Patient's basic information*

⇒ *includes**:

{
- *Age*
- *Gender*
- *Stage of Disease*
- *Educational Background*
- *Language Habits*
- *Health Status*

}

Patient Voice Characteristics Ontology clearly defines the key features of patient voice data (e.g., pitch, volume, rhythm, speech clarity), recording environment parameters, and basic patient information. Based on this ontology, hardware can be configured accordingly in terms of fixed-point/floating-point formats, register bit-widths, and estimated data throughput. Since the ontology specifies the semantics and units of each feature, hardware and software components such as drivers and register mappings can adopt consistent field identifiers, thereby avoiding issues related to data misalignment or naming conflicts.

Based on the clear definition of voice data characteristics, the "Voice Sound Processing Rules Ontology" delineates the entire pipeline of voice data processing—from acquisition, segmentation, and denoising to feature extraction, algorithmic analysis, and result validation. Through the key algorithm identifiers within the ontology, the hardware layer can pinpoint the functional modules to be implemented and assess the resource requirements of each algorithm. This ontology not only directly guides the selection of algorithms and the design of functional modules necessary for hardware acceleration, but also reserves a well-defined semantic space and data interfaces for future system extensions, such as the integration of novel machine learning algorithms or emotion analysis modules.

***Voice Sound Processing Rules Ontology***

:= [Describes the specific rules and methods for processing and analyzing patient voice data.]
:= [Focused on the procedures and rules for processing sound data, with the aim of ensuring consistency and accuracy in data analysis.]
⇒ *subdividing**:
{
- *Feature Extraction Methods*
  - := [How to extract useful information or features from raw sound data.]
  - ⇒ *includes**:
    {
    - *Spectral Analysis*
    - *Time-Domain Analysis*
    - *Cepstral Analysis*
    - *Morphological Feature Extraction*

    }
- *Voice Analysis Algorithms*
  - := [Analyze these features to identify patterns related to diseases.]
  - ⇒ *includes**:
    {
    - *Machine Learning Algorithms*
      - ⇒ *includes**:
        {
        - *SVM*
        - *Random Forest*
        - *Deep Neural Networks*
          - ∋ *CNN*
        - *LSTM*
        - *PCA*

        }
    - *Voice Biomarker Identification*
    - *Emotion Analysis*

    }
- *Data Cleaning and Preprocessing*
  - := [Ensure data quality]
  - ⇒ *includes**:
    {
    - *Noise Reduction*
    - *Voice Segmentation*
    - *Normalization*

    }
- *Evaluation and Validation*
  - := [Ensure the effectiveness and reliability of the analysis methods]
  - ⇒ *includes**:
    {
    - *Cross-Validation*
    - *Performance Metrics*
      - ⇒ *includes**:
        {
        - *Accuracy*
        - *Recall*
        - *Precision*
        - *F1 Score*
        - *ROC Curve*
        - *AUC*

        }

    }

}

## III. Proposed Approach

### A. Development of the Solver and Knowledge Base.

A knowledge base is a broad term referring to an electronic repository for storing knowledge information, which can be structured or unstructured, including text documents, databases, images, videos, etc. Figure 1 illustrates the complete knowledge engineering process from concept collection and requirements validation to ontology construction, knowledge base application, and maintenance, realizing end-to-end modeling of "domain knowledge to hardware-driven execution" within an intelligent system. The process includes four main phases: Coordination Phase, Establishment Phase, Application Phase, and Maintenance Phase.

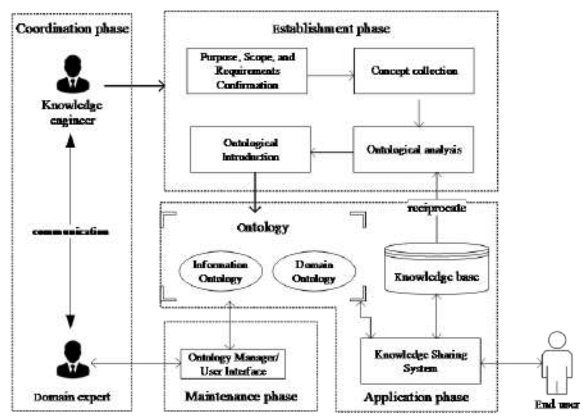


Figure 1. Ontological Engineering Architecture Diagram

The Coordination Phase emphasizes continuous communication between knowledge engineers and domain experts. Knowledge engineers are responsible for formalizing the experts' knowledge into ontologies, while domain experts provide core knowledge such as diagnostic experience, domain terminology, and data usage logic.

The Establishment Phase is the core of the process, forming the main development workflow for ontologies. It defines the objectives of ontology construction (e.g., for AD voice diagnostics), its scope (e.g., covering voice and semantics, excluding vision or behavior), and specific technical/data/interaction requirements. Based on domain expert input and literature review, core concepts related to the modeling objective are systematically collected and modeled—such as "voice features," "pause duration," and "FPGA feature extraction module." The collected concepts undergo structural analysis to identify their relationships (e.g., is-a, part-of, used-by, measured-in), and a hierarchical concept structure is built. The constructed ontology structure is then imported into the system to interface with the knowledge base and other applications. This stage can be seen as a "coding" phase, where ontological concepts are represented in a formal, machine-readable format (e.g., SC-code).

The final output of the Establishment Phase is the Ontology, which comprises two subclasses: Information Ontology and Domain Ontology. Together, these ontologies form the semantic backbone of the entire knowledge architecture—interfacing downward with the knowledge base and upward with diagnostic logic, rules, and interaction definitions.

Ontologies and knowledge bases are reciprocate—ontologies define the semantic structures for the knowledge base, while actual data in the knowledge base can feed back into the refinement and expansion of the conceptual system. In the knowledge base storage system, concept instances defined by the ontology correspond to real data, such as:

1) Structured voice feature data (e. g., pitch=150Hz, pause_duration=0.7s);
2) Hardware configuration files (e. g., FPGA register allocation, interface specifications);
3) Medical records, expert diagnostic criteria, etc.

The Application Phase represents the front-end where end users interact with the system, such as doctor terminals, hardware monitoring interfaces, and AD diagnostic dashboards. It retrieves the necessary inference information from the knowledge base and provides intelligent assistance to hardware algorithm modules, diagnostic engines, or clinical interfaces.

The system also generates new data and rules (e. g., high-risk features newly identified by models), which are fed back into the knowledge base, forming a cycle of knowledge enrichment.

The Maintenance Phase allows knowledge engineers or technical experts to perform dynamic updates, offering graphical interfaces or toolkits for maintaining the ontology structure, updating conceptual definitions, and fixing semantic errors, thus adapting to evolving medical standards, hardware changes, or new modules.

The foundation of the ostis-system problem solver is a set of agents (sc-agents) that interact solely through information processing rules executed within the semantic memory [9]. As a multidimensional, multilingual, layered, and nestable collection of sc-agents, it provides a clear hierarchical structure and flexible nesting mechanism, enabling seamless alignment from high-level functional requirements to low-level executable procedures, ensuring efficient, flexible, and scalable complex problem-solving capabilities.

The following problem solver description outlines how processes such as feature extraction, reasoning, and result synthesis are accelerated using FPGA hardware, and how the hardware interacts with the ontology and knowledge base.

***FPGA Problem Solver for AD Diagnostic System***
:= [sc-mode of the ostis-system problem solver with hardware acceleration]
⇒ *Decomposition**:
{• *sc-agent for input data acquisition*
⇒ *Decomposition**:
{• *sc-agent for identifying data type*
• *sc-agent for receiving raw data*
• *sc-agent for synchronizing with the hardware interface*
• *sc-agent for storing raw data into the knowledge base*
}
• *sc-agent for FPGA-accelerated data processing*
⇒ *Decomposition**:
{• *sc-agent for data preprocessing*
⇒ *Decomposition**:
{• *sc-agent for noise reduction*
• *sc-agent for tokenization/segmentation*
• *sc-agent for basic transformations*
}
• *sc-agent for feature extraction*
⇒ *Decomposition**:
{• *sc-agent for TF-IDF computation*
• *sc-agent for dimensionality reduction*
• *sc-agent for fixed-point quantization / converting data format for FPGA*
}
• *sc-agent for FPGA pipeline control*
⇒ *Decomposition**:
{• *sc-agent for uploading model parameters*
• *sc-agent for running inference on FPGA*
• *sc-agent for reading back results*
• *sc-agent for optimizing FPGA resource usage*
}
• *sc-agent for storing intermediate results*
}
• *sc-agent for inference and reasoning*
⇒ *Decomposition**:

{• *sc-agent for random forest voting*
• *sc-agent for comparing results with knowledge base thresholds*
• *sc-agent for final classification (AD / non-AD)*
}
• *sc-agent for results output*
⇒ *Decomposition**:
{• *sc-agent for retrieving classification outcome from the knowledge base*
• *sc-agent for generating explanation*
• *sc-agent for natural language generation / UI display*
}
}

*B. Designing the User interface.*

In terms of interactive interface design, during the data presentation stage of the diagnostic network, referring to article [10], the authors considered factors such as the user interface concept within the OSTIS ecosystem and adopted a component-based approach to design an adaptive intelligent multimodal interface for the OSTIS system.

At the current stage, OSTIS primarily functions as a 'semantic web/knowledge graph visualization editor,' allowing users to draw concepts, define relationships, and perform basic reasoning operations within the platform. The logical design concept of this interactive interface is described in scn-code as follows:

***Hardware-Supported Diagnostic Interface for AD***
:= [HCI page that allows real-time data acquisition from FPGA, performs hardware-accelerated inference, and displays final results]
⇒ *Decomposition**:
{• *Non-atomic UI components*
⇒ *Example**: *Inclusion**:
{• *Hardware Control Panel*
⇒ *Example**: *Inclusion**:
{• *Start Hardware Button*
• *Stop Hardware Button*
• *Config Hardware Button*
• *Hardware Status Indicator*
}
}
• *Data Acquisition & Monitoring Panel*
⇒ *Example**: *Inclusion**:
{• *Audio/Voice Input Button*
• *Realtime Waveform Display*
• *FPGA Processing Throughput Indicator*
}
• *Diagnostic Inference Panel*
⇒ *Example**: *Inclusion**:
{• *Predict AD Button*
• *Show Confidence Score Checkbox*
• *Additional Explanation Toggle*
}
• *Knowledge Base / Ontology Access Panel*
⇒ *Example**: *Inclusion**:
{• *Query Existing Records Button*
• *Show Concept Graph Button*
• *Export Results to Knowledge Base Button*
}
• *Atomic UI components*
⇒ *Example**: *Inclusion**:
{• *Logs & Alerts Area*
• *Final Diagnosis Display Area*
}
}

## IV. FPGA accelerator

In practical implementation, the authors proposed an FPGA-based hardware accelerator aimed at addressing the performance bottlenecks of random forest classification models in intelligent text analysis tasks when executed on traditional CPU and GPU platforms. The FPGA accelerator is specifically designed to perform inference of the random forest algorithm, targeting scenarios characterized by constrained resources, real-time requirements, and low power consumption, such as those found in edge computing environments.

To effectively adapt to the resource limitations inherent to FPGA platforms, the authors applied a lightweight algorithmic design strategy on the hardware side. This included significant dimensionality reduction of feature vectors as well as quantization of floating-point values into fixed-point representations. These design choices reduced the consumption of FPGA logic resources and improved computational efficiency by eliminating the need for costly floating-point arithmetic units.

Regarding architectural design, the FPGA accelerator employs a combination of parallelism and pipelining. Each decision tree within the random forest is implemented in parallel on the FPGA, enabling independent execution of inference operations. The final classification result is obtained through a majority voting scheme. Furthermore, the inference process within each tree is pipelined, with threshold comparisons and feature evaluations conducted using efficient fixed-point operations, thereby further reducing latency.

During hardware implementation, the authors utilized high-level synthesis directives provided by the Vitis HLS development environment—such as #pragma HLS PIPELINE and #pragma HLS UNROLL—to automate pipelining and enhance intra-module parallelism. The

FPGA communicates with the CPU via AXI-Lite or memory-mapped interfaces, simplifying data exchange and facilitating system integration between hardware and software components.

The study also devoted attention to evaluating resource utilization and performance of the implemented accelerator. Synthesis results showed moderate consumption of logic elements (LUTs), flip-flops (FFs), and DSP blocks, with sufficient remaining resources available on the FPGA for potential future expansions. Additionally, performance measurements conducted in the Vitis HLS environment revealed a theoretical acceleration ratio of approximately 427,840×. However, this figure represents an idealized estimate, as it compares a simulated software environment with pure hardware execution and does not account for practical factors such as I/O latency.

## V. Conclusion and future works

In the report, the authors integrate the OSTIS «semantic layer—knowledge base—problem solver» hybrid intelligent approach with hardware resources such as FPGA, introducing a novel technical framework characterized by high efficiency, low latency, traceability, and scalability for Alzheimer's disease diagnostic scenarios. The described "hardware + software" pipelined collaborative model emphasizes interaction between the ostis-system semantic framework and sc-agent components. For practical engineering implementation, each sc-agent can be regarded as a software task module running on the CPU side, paired with parallel computing or inference units on the FPGA, thereby constructing a hybrid pipeline from data acquisition to inference output. Future research directions may focus on further algorithmic optimization, expanded multimodal data integration, and large-scale clinical validation to reinforce the clinical effectiveness and applicability of the proposed system.

## References


[1] Parul, A. Singh, and S. Shukla, "Novel techniques for early diagnosis and monitoring of alzheimer's disease," *Expert Review of Neurotherapeutics*, vol. 25, no. 1, pp. 29–42, 2025.

[2] (2024) Alzheimer's disease facts and figures. [Online]. Available: https://www.alz.org/alzheimers-dementia/facts-figures

[3] I. Martínez-Nicolás, T. E. Llorente, F. Martínez-Sánchez, and J. J. G. Meilán, "Ten years of research on automatic voice and speech analysis of people with alzheimer's disease and mild cognitive impairment: A systematic review article," *Frontiers in Psychology*, vol. 12, p. 620251, 2021. [Online]. Available: https://doi.org/10.3389/fpsyg.2021.620251

[4] Y. Tang, X. Xiong, G. Tong, Y. Yang, and H. Zhang, "Multimodal diagnosis model of alzheimer's disease based on improved transformer," *BioMedical Engineering OnLine*, vol. 23, no. 8, 2024. [Online]. Available: https://doi.org/10.1186/s12938-024-01204-4

[5] V. U. Anatol'evich and C. Yu, "Designing an iot network for the diagnosis of alzheimer's disease using ostis," in *Open Semantic Technologies for Intelligent Systems (OSTIS): Proceedings of Scientific Works*, V. V. Golenkov *et al.*, Eds., vol. 8. Minsk: Belarusian State University of Informatics and Radioelectronics, 2024, pp. 243–248.

[6] A. Abdulkadir, B. Mortamet, P. Vemuri, C. R. Jack, G. Krueger, and S. Klöppel, "Effects of hardware heterogeneity on the performance of svm alzheimer's disease classifier," *NeuroImage*, vol. 58, no. 3, pp. 785–792, 2011.

[7] H. Qin and Q. Cao, "Design of convolutional neural networks hardware acceleration based on fpga," *Journal of Electronics & Information Technology*, vol. 41, no. 11, pp. 2599–2605, 2019.

[8] V. Krischenovich, D. Salnikov, and V. Zahariev, "Integration and standardization in new generation intelligent medical systems based on ostis technology," in *Open Semantic Technologies for Intelligent Systems (OSTIS): Proceedings of Scientific Works*, V. V. Golenkov *et al.*, Eds., vol. 8. Minsk: Belarusian State University of Informatics and Radioelectronics, 2024, pp. 157–164.

[9] D. Shunkevich, "Principles of decentralized problem solving within the ecosystem of next-generation intelligent computer systems," in *Open Semantic Technologies for Intelligent Systems (OSTIS): Proceedings of Scientific Works*, V. V. Golenkov *et al.*, Eds., vol. 8. Minsk: Belarusian State University of Informatics and Radioelectronics, 2024, pp. 15–28.

[10] M. Sadouski, "User interface of the ostis ecosystem," in *Open Semantic Technologies for Intelligent Systems (OSTIS): Proceedings of Scientific Works*, V. V. Golenkov *et al.*, Eds., vol. 7. Minsk: Belarusian State University of Informatics and Radioelectronics, 2023, pp. 153–158.


# АППАРАТНАЯ ПОДДЕРЖКА В ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЕ ИТ-ДИАГНОСТИКИ БОЛЕЗНИ АЛЬЦГЕЙМЕРА


Вишняков В.А., Юй Чуюэ



В докладе представлен интегрированный интеллектуальный диагностический фреймворк, сочетающий семантически ориентированное представление знаний на основе онтологий, аппаратное ускорение с использованием FPGA и концепции OSTIS с целью повышения эффективности обработки в реальном времени для ИТ-диагностики пациентов с болезнью Айцгеймера. Разработана онтология для таких пациентов, на основании которой представлены база знаний, решатель и интеллектуальный интерфейс. Предложен FPGA-ускоритель, за счет которого интеллектуальная система ИТ-диагностики пациентов с БА обеспечивает эффективный сбор, предварительную обработку, извлечению признаков и семантический анализ голосовых данных, что позволяет сократить задержки в диагностике и повысить оперативность принятия решений, примерно на порядок. Разработанные онтологии согласованы с семантическими требованиями концепции OSTIS. Разработанная гибридная архитектура CPU-FPGA, и предложенный пользовательский интерфейс интегрирует запросы к базе знаний, управление аппаратными средствами, мониторинг в реальном времени и диагностическое заключение, демонстрируя улучшенное взаимодействие с пользователем, семантическую прозрачность и операционную эффективность.



Received 28.03.2025