

UNICA UNIVERSITÀ DEGLI STUDI DI CAGLIARI



This is the Author's *submitted* manuscript version of the following contribution:
Luigi Serreli, Alessandro Pruner, Luigi Atzori, Michele Nitti,
Giving voice to digital twins: How LLMs build human knowledge graphs,
Computer Communications, Volume 255 (2026), art. 108553.

The publisher's version is available at:
https://doi.org/10.1016/j.comcom.2026.108553

When citing, please refer to the published version.

© 2026. This manuscript version is made available under the CC-BY-NC-ND 4.0 license https://creativecommons.org/licenses/by-nc-nd/4.0/

This full text was downloaded from UNICA IRIS https://iris.unica.it/

Giving Voice to Digital Twins: How LLMs Build Human Knowledge Graphs

Luigi Serreli[a,b], Alessandro Pruner[a], Luigi Atzori[a,b], Michele Nitti[a,b]

[a]*Department of Electrical and Electronic Engineering, University of Cagliari, Via Marengo 2, Cagliari, 09123, , Italy*
[b]*National Telecommunication Inter University Consortium, Research Unit of Cagliari, Via Marengo 2, Cagliari, 09123, , Italy*

Abstract

The convergence of the Internet of Things, artificial intelligence, and semantic technologies is reshaping how digital systems model and reason about human behavior. Human Digital Twins (HDTs) offer dynamic user representations by integrating multimodal data from physiological, behavioral, and linguistic sources. However, most existing HDTs perform low-level data fusion, lacking the semantic consistency needed for coherent reasoning and personalization. This paper proposes a novel pipeline for constructing Personal Knowledge Graphs (PKGs), i.e. adaptive, formal representations of user knowledge, that combine the generative capabilities of Large Language Models (LLMs) with ontology-based validation. The pipeline includes triplet extraction from language data, semantic alignment through embeddings, and ontology-driven graph construction. A comparative analysis between ontology-free and ontology-guided PKG generation demonstrates that semantic grounding substantially reduces spurious relationships and improves reasoning accuracy, establishing ontological validation as essential for reliable knowledge extraction. By bridging symbolic and generative paradigms, the proposed approach advances knowledge-driven HDTs that are proactive, interpretable, and resilient.

Keywords: Human Digital twins, Personal Knowledge Graph, Large Language Models, Ontology

1. Introduction

The convergence of the Internet of Things (IoT), artificial intelligence (AI), and semantic modeling is enabling systems capable of representing human behavior through continuous, multimodal data assimilation. Within this landscape, Human Digital Twins (HDTs) have gained attention as dynamic, user-centric digital counterparts built from physiological, behavioral, and linguistic information [1]. Through such integration, HDTs promise a shift from static monitoring toward proactive modeling of user needs and preferences, bridging sensing, reasoning, and personalization [2].

Despite these advancements, most HDT implementations rely on low-level data fusion or application-specific modeling, often overlooking the semantic alignment needed for consistent knowledge representation across heterogeneous sources [3]. Physiological sensors capture objective signals, while language interfaces express subjective intentions, yet few architectures integrate these streams within a unified, ontology-driven framework. As a result, systems frequently lack cognitive coherence and struggle to infer cross-modal relationships or explain user behavior [4].

Recent research suggests that combining ontological reasoning with Large Language Models (LLMs) provides a powerful foundation for overcoming these limitations [5]. Ontologies supply formal structures for entities and relations, ensuring interpretability and reusability across contexts. LLMs, conversely, enable extraction of structured information from unstructured discourse, supporting the transformation of linguistic content into semantically enriched representations. Their integration allows the construction of Personal Knowledge Graphs (PKGs), i.e. formal, dynamic, user-specific representations that serve as the cognitive substrate of HDTs [6]. However, combining symbolic and generative reasoning remains challenging: LLMs may introduce hallucinated or ontologically inconsistent relations, whereas symbolic systems lack flexibility in handling open-ended human input. The practical importance of semantically grounded HDTs spans diverse domains, from autonomous digital representatives maintaining social presence in virtual environments [7] to health monitoring systems integrating physiological signals with linguistic symptom descriptions. Without formal semantic validation, extracted relationships risk introducing inconsistencies that compromise reliability in decision-critical applications.

This work addresses these challenges by proposing a knowledge-driven pipeline for constructing PKGs within HDT architectures. The proposed framework defines how heterogeneous observations about a user can be transformed into a structured and semantically grounded knowledge representation capable of supporting proactive reasoning. The pipeline operationalizes the concept of a Personal Digital Brain (PDB) as a semantic coordination layer that transforms heterogeneous data streams, including conversational utterances, physiological measurements, calendar events, and environmental observations, into validated, ontology-aligned knowledge representations [8]. The interaction between linguistic extraction, multimodal contextualization, and ontology-based validation ensures that extracted relationships are both expressive and semantically coherent.

In summary, the proposed PKG construction pipeline is implemented and validated through three key contributions:

- A three-stage triple acquisition mechanism that integrates linguistic expressions with multimodal observations

through the Model Context Protocol, enabling temporal alignment between conversational utterances and heterogeneous data sources.

- An ontology-based semantic alignment mechanism employing parallel validation strategies that propagate type constraints through formal ontological structures, ensuring extracted knowledge adheres to domain semantics while maintaining robustness to extraction errors.
- A comparative experimental analysis showing that semantic grounding substantially improves reasoning accuracy and reduces spurious relationships, establishing ontology-driven validation as a fundamental requirement for conversational knowledge extraction in HDT architectures.

The remainder of the paper is organized as follows. Section 2 reviews the state of the art in HDT architectures, conversational knowledge extraction, and ontological grounding mechanisms. Section 3 presents the system architecture, while Section 4 details the Digital Brain pipeline, including triple acquisition, semantic alignment, and PKG construction. Section 5 describes the simulation setup and experimental methodology and Section 6 reports the evaluation results. Finally, Section 8 concludes the paper with a discussion of the implications and future research directions.

2. State of Art

The development of HDT systems capable of constructing comprehensive PKGs requires addressing three interconnected research domains. We first examine existing HDT architectures to identify their structural capabilities and limitations in handling multimodal data integration. We then analyze recent advances in knowledge extraction from conversational interactions, focusing on how ML models transform unstructured linguistic content into structured semantic representations and the challenges of maintaining semantic consistency during this transformation. Finally, we position our work within the broader trajectory from passive data monitoring toward proactive computational modeling, examining how ontology-driven approaches enable the transition from digital shadows to genuine digital twins capable of inferential reasoning.

2.1. Digital Twin Architectures for Personal Data

Digital Twins (DTs) were originally designed for monitoring, simulation, prediction, and decision-making/control in engineered systems [9]. In these domains, physical processes can be modeled through measurable and well-structured parameters, allowing digital replicas to support forecasting and automated control loops.

Extending this paradigm to humans is considerably more complex. Modern platforms collect heterogeneous data, such as physiological signals, behavioral logs, and conversational content, through wearables, mobile applications, and natural language interfaces [10]. Yet, most current implementations function as isolated data collectors, lacking the semantic integration required for holistic interpretation. In the context of HDTs, the digital twin itself represents the entity responsible for aggregating heterogeneous observations about the user, including sensor measurements, application data, and contextual information. This unified representation enables different services within the HDT to access and reason over user-related observations. In our framework, the Digital Brain is introduced as a service operating within the HDT that transforms these aggregated observations into structured knowledge through the proposed PKG construction pipeline. By enabling this knowledge-driven integration, the HDT can evolve beyond traditional monitoring and prediction functionalities toward supporting higher-level capabilities such as decision optimization and proactive responses. Indeed, sensor streams often miss temporal and causal context, application logs capture only discrete events, and conversational data remains unstructured, preventing unified representations suitable for inference or prediction.

HDTs aim to overcome these limitations by shifting from passive observation toward dynamic computational modeling of individuals [11]. However, despite advances in multimodal sensing, existing solutions largely remain confined to monitoring and limited prediction, while the higher-level DT functions, i.e. decision support and proactive control, remain mostly unrealized for human-centered scenarios. This gap reflects the intrinsic difficulty of modeling human behavior, which emerges from the interaction of physiological, cognitive, emotional, and contextual factors [12].

To progress beyond monitoring, HDTs must integrate three capabilities identified in prior work. Multimodal synthesis must unify structured parametric measurements with unstructured linguistic inputs and contextual event streams, creating coherent representations that capture complementary aspects of human states [13]. Semantic integration through formal knowledge frameworks transforms disparate data sources into machine-interpretable structures, ensuring that physiological readings, behavioral patterns, and verbal expressions can be reasoned upon within consistent ontological models [14]. Inferential reasoning mechanisms leverage these integrated representations to perform predictive simulations, identify temporal patterns, and detect conditions that manifest through cross-modal correlations rather than isolated measurements [15].

These requirements highlight the need for formal knowledge representation mechanisms, in the form of Personal Knowledge Graphs, which must be capable of organizing multimodal observations into coherent, queryable structures and leveraging the rich semantic content conveyed by conversational interactions. Recent research has also emphasized the importance of privacy-preserving mechanisms and distributed infrastructures when deploying HDT systems in healthcare and IoT environments. Since HDTs continuously process sensitive personal data such as physiological signals, behavioral traces, and conversational information, several works have investigated privacy-aware learning and system architectures. For instance, the framework proposed in [16] introduces a differentially private federated multi-task learning approach that enables collaborative model optimization across distributed HDT nodes while protecting sensitive user data from direct exposure. Complementary perspectives on the technological ecosystem required

for HDT deployment are provided by recent surveys such as [17] and [18]. These works analyze the networking architectures, enabling technologies, and AI-driven components that support HDT systems in personalized healthcare and IoT environments, highlighting key challenges related to large-scale multimodal data acquisition, distributed computation, privacy preservation, and intelligent data analysis. These studies further confirm the importance of integrating semantic modeling, multimodal reasoning, and privacy-aware infrastructures in future HDT architectures.

2.2. *Personal Knowledge Graphs and Conversational Extraction*

Conversational interactions constitute the richest source of semantic information for HDTs, as they convey user states, intentions, preferences, and contextual experiences [19] [20]. Unlike physiological signals or application logs, which provide structured measurements, linguistic expressions encode meaning through flexible constructions, pragmatic cues, and contextual dependencies [21]. This information is not explicitly represented: semantic relationships are often implicit [22], and utterances may rely on dialogue history, anaphora, or elliptical forms that require inferential interpretation [23] [24]. These properties make conversational content valuable for human modeling but difficult to transform into formal representations suitable for PKG construction.

Recent advances in LLMs have significantly improved this process. Traditional discriminative approaches required domain-specific annotated datasets and were limited in generalization [25]. Modern LLMs instead generate structured outputs directly from natural language instructions, enabling zero-shot and few-shot extraction [26]. Through prompting strategies such as format-specific instructions and guided reasoning, models learn to produce subject–predicate–object triples aligned with the intended schema [27]. Empirical studies demonstrate substantial gains in accuracy, with improvements attributed to techniques such as in-context learning [28] and chain-of-thought prompting [29]. Precision above 50% for complete triple extraction from dialogue has been reported in [30], and comparative analyses show LLM-based systems achieving F1 scores around 0.69, outperforming traditional extractors [31].

However, several limitations remain critical for HDT knowledge graph construction. LLMs continue to hallucinate facts, generating incorrect triples with error rates between 5% and 10% [32]. They may produce inconsistent or non-standardized terminology, yielding vague entities or relations that lack semantic specificity [33]. Domain-specific evaluations show sharp performance drops for models not explicitly trained on expert corpora, for example in biomedical extraction tasks [34]. Furthermore, LLM outputs generally lack grounding: surface forms may refer to the same concept with different expressions, violate type constraints, or create ambiguous entities without canonical identifiers.

In HDT contexts, conversational extraction also requires integration with structured contextual information from heterogeneous data sources. The meaning of an utterance often depends on concurrent physiological signals, temporal conditions, or environmental events not explicitly mentioned by the user [35]. Emerging frameworks such as the Model Context Protocol provide unified access to such multimodal information, enabling LLMs to incorporate external context during reasoning [36] [37]. Nonetheless, their systematic application to temporally aligned PKG construction remains underexplored.

These limitations highlight the need for validation and grounding mechanisms capable of reconciling probabilistic LLM outputs with deterministic ontological structures before integration into PKGs. Without this step, early extraction errors or semantic inconsistencies risk propagating through the longitudinal HDT representation, undermining the reliability of downstream reasoning. Ontology-based grounding therefore becomes essential for ensuring coherence, canonicalization, and confidence-aware integration of extracted triples, a foundation addressed in the following subsection.

2.3. *Ontological Grounding for Proactive Digital Twins*

The limitations identified in conversational knowledge extraction necessitate robust validation mechanisms capable of grounding LLM-generated triples within formal semantic structures before integration into Personal Knowledge Graphs. While LLMs demonstrate remarkable extraction capabilities, their outputs lack the semantic formalization required for reliable knowledge representation. Extracted triples consist of natural language strings that carry ambiguity, may violate domain constraints, and often employ non-standardized terminology inconsistent with established ontological frameworks. Ontologies provide formal specifications of domain concepts, their properties, and the constraints governing their relationships, offering structured frameworks against which extracted knowledge can be validated [27]. A key challenge is reconciling probabilistic neural outputs with deterministic ontological constraints, ensuring that extracted entities and relations are mapped to canonical concepts with quantified confidence.

Existing approaches to ontology-based validation address schema-level integration rather than instance-level validation during conversational extraction. Ontology alignment methodologies have evolved from lexical similarity measures toward embedding-based approaches that capture semantic relationships in continuous vector spaces [38, 39], with techniques such as OWL2Vec incorporating graph structure through random walks and recent work exploring LLM-based matching [40, 41]. However, these methods perform batch processing between complete ontologies, incompatible with real-time validation of individual triples from streaming conversational data. Semantic Web validation frameworks such as SHACL and OWL reasoning provide constraint checking over RDF graphs [42], but assume pre-grounded inputs with entities already mapped to ontological URIs, making them unsuitable for raw string-based triples from LLMs. Complementary embedding-based approaches like TransE and DistMult enable plausibility scoring through learned vector representations [43], while neurosymbolic frameworks integrate neural learning with logical constraints [44], yet these function as post-hoc reasoning engines for knowledge graph completion rather than pre-admission validation mechanisms.

These limitations reflect a broader pattern in HDT literature where implementations function as digital shadows that passively replicate observations rather than digital twins capable of proactive modeling. Without semantic grounding, systems accumulate multimodal data but lack the formal knowledge representations necessary for inferential reasoning, temporal pattern detection, and causal inference. No existing framework simultaneously addresses the requirements for conversational HDT with proactive capabilities: instance-level processing of individual triples during incremental extraction, confidence-scored validation quantifying semantic fit for uncertain LLM outputs rather than binary acceptance, latency for interactive dialogue, integration of multimodal context through protocols like MCP, and ontological grounding that enables cross-modal reasoning. The alignment mechanism presented in Section 4 addresses these gaps by combining embedding similarity with signature propagation through ontological constraints, enabling the construction of semantically coherent PKGs that support genuine digital twin functionality.

2.4. *Comparative Analysis and Positioning of the Proposed HDT*

While the previous subsections have analyzed the architectural, extraction, and grounding challenges underlying HDT systems, it is equally important to position the proposed framework with respect to recent digital twin implementations that integrate knowledge graphs, workflow orchestration, or task optimization mechanisms.

Recent research has explored digital twin architectures primarily from the perspective of infrastructure optimization and service orchestration. SerFlow [45] proposes a multi-stage service-enhanced mechanism for workflow applications in cyber-physical systems, emphasizing end–edge–cloud collaboration and execution efficiency. Although it advances distributed orchestration strategies, its focus remains on computational workflow performance rather than semantic knowledge construction or conversational grounding. Similarly, the digital twin-enabled dynamic spatial–temporal knowledge graph introduced in [46] leverages IoT data and graph reasoning for production logistics resource allocation. This approach demonstrates the effectiveness of integrating spatial and temporal modeling with knowledge graph structures, yet its objective lies in industrial resource optimization rather than longitudinal human-centered modeling.

In the socio-cyber domain, TOFDS [47] presents a two-stage task execution method based on reinforcement learning and digital twin simulation for fake news processing. While it incorporates optimization and dynamic decision-making within a digital twin paradigm, it does not address semantic grounding of conversational knowledge or personal knowledge graph construction. More recently, the knowledge graph-based data integration system for digital twins of built assets [48] emphasizes modularity and interoperability across Building Information Modeling (BIM), IoT, and process-level data. This work advances semantic data integration for infrastructure-scale twins but remains oriented toward asset management rather than proactive cognitive modeling.

Table 1: Comparison of digital twin approaches (KG: Knowledge Graph, Opt.: Optimization, H-PKG: Human-centric PKG).

Work	**Focus**	**KG**	**Opt.**	**H-PKG**
SerFlow [45]	CPS workflow	No	Yes	No
DSTKG [46]	Logistics	Yes	Yes	No
TOFDS [47]	Socio-cyber	No	Yes	No
KGDT [48]	Built assets	Yes	No	No
Proposed HDT	Personal DT	Yes	Partial	Yes

To clarify these distinctions, Table 1 summarizes the core characteristics of these approaches and contrasts them with the proposed HDT framework.

Unlike infrastructure-oriented digital twins that primarily optimize physical processes or service execution, the proposed HDT framework focuses on the construction and validation of a longitudinal PKG grounded in conversational and multimodal data. Its novelty lies in the instance-level ontological validation of probabilistic LLM outputs before graph integration, enabling proactive inferential reasoning rather than passive data replication. In this sense, the proposed architecture advances the transition from digital shadows to semantically grounded digital twins capable of structured human-centered modeling.

3. System Architecture

This section presents the architectural framework that enables the construction of PKGs through coordinated integration of heterogeneous data sources. The architecture transforms raw measurements from sensors and applications into semantically grounded knowledge representations suitable for proactive reasoning. We detail how the system acquires multimodal observations, processes them through domain-specific services, and organizes validated relationships within the PKG maintained by the HDT. The proposed architecture follows a user-centric design in which each individual is associated with a dedicated HDT. In multi-user environments, multiple HDTs can coexist independently, each operating within its own user space and maintaining an independent knowledge representation through its PKG.

Figure 1 illustrates the proposed architecture, organized into three interconnected layers that progressively convert observations into structured user representations. The HDT continuously receives heterogeneous observations related to the user and the surrounding environment. These observations may originate from multiple sources, including physical-world objects (e.g., microphones and IoT devices) as well as digital services and applications used by the user. Such observations may include both verbal information, such as conversational utterances, and non-verbal contextual signals, such as environmental measurements, physiological indicators, or scheduled activities. These heterogeneous inputs are progressively integrated across the architectural layers illustrated in Figure 1.

Communication across layers is supported by the Model Context Protocol (Section 3.2.1), which provides unified interfaces for accessing measurements, events, and processed ob-

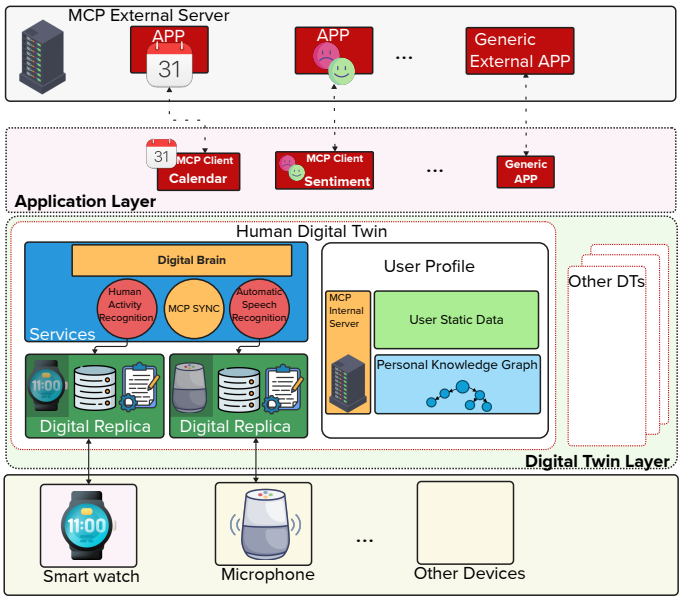


Figure 1: HDT System Model architecture

servations generated by the different components of the system. Within this architecture, the Digital Brain acts as the coordination layer that aggregates the available observations and organizes them into structured representations of the user. These representations are then processed through the knowledge construction pipeline described in the following sections, which converts heterogeneous inputs into semantically validated knowledge stored in the Personal Knowledge Graph.

In the following subsections, we describe the main architectural components and their role in supporting the proposed HDT framework.

3.1. Digital Twin Layer

The DT Layer instantiates digital representations of physical entities that continuously receive measurements, process them through computational services, and maintain synchronized state with their physical counterparts. Each DT performs active processing rather than simple data replication, applying domain-specific algorithms to transform raw sensor readings into outputs tailored to the entity's sensing characteristics.

The architecture supports heterogeneous sensing environments containing multiple physical entities operating simultaneously. Let $E = e_1, e_2, \ldots, e_{|E|}$ denote the set of physical entities, where each e_i corresponds to a sensing device such as a wearable for physiological monitoring, an environmental sensor, or an audio interface. Each entity provides distinct sensing modalities and temporal sampling characteristics determined by its embedded instrumentation.

For each entity $e_i \in E$, the system instantiates one or more DTs that receive, process, and expose the corresponding measurements. Every DT consists of three interconnected components: digital replicas, services, and profile.

Digital replicas maintain synchronized representations of the physical entity's current state through bidirectional communication. Replicas continuously receive measurements and can propagate commands or configurations back to the device. A measurement produced by entity e_i is represented as:

$$m^j_{e_i} = (a^j_{e_i}, v^j_{e_i}, \tau^j_{e_i}) \quad (1)$$

where $a^j_{e_i}$ denotes the parameter type (e.g., temperature, humidity), $v^j_{e_i}$ the observed value, and $\tau^j_{e_i}$ the acquisition timestamp. For each type j, $a^j_{e_i}$ is fixed, while the entity generates a temporal sequence of measurements incoming with varying values and timestamps. The digital replica stores the most recent state, enabling immediate access without querying the physical hardware.

Services implement the processing layer that applies computational algorithms to the measurements stored in the replica. These modular units execute domain-specific processing, such as signal filtering for physiological sensors, pattern recognition for motion data, temporal aggregations, or feature extraction from raw streams. Services expose their processed outputs through standardized interfaces, allowing higher-layer applications to consume derived features without handling device-specific logic.

Profile stores the persistent metadata and static characteristics of the physical entity, including sensing capabilities, configuration parameters, calibration data, and other invariant attributes. While the digital replica maintains a dynamic measurement state, the profile captures the static information required to correctly interpret and process measurements.

This three-component structure defines the general architecture for all DTs in the system, applicable across domains from networked IoT devices [49] to human modeling. Within this framework, the HDT constitutes a specialized instantiation: it retains the same structure but extends the Profile into a User Profile that captures both static user characteristics and the dynamic PKG that evolves through continuous knowledge integration.

3.1.1. Human Digital Twin

The HDT is a specialized DT designed to model a human being. Unlike other DTs, it aims to capture the complexity of human presence, activities, state, and knowledge. This requirement shapes its architecture, particularly the User Profile, which extends beyond static metadata to include dynamic knowledge structures maintained in the PKG.

The HDT contains multiple digital replicas, each corresponding to a physical device associated with the user. Wearables, audio interfaces, and other personal devices each maintain their own replica, continuously receiving measurements from their physical sources. This multi-replica design reflects the need to aggregate heterogeneous sensing modalities to achieve comprehensive user modeling. Each replica follows the general DT replication model: it maintains synchronized state and exposes measurements to the HDT's service layer.

The service layer follows a hierarchical organization. Base services process measurements from individual replicas using modality-specific algorithms, such as activity classification from motion data, physiological indicators from biosensors, or

transcription from audio streams. These services adhere to the same architectural principles as generic DT services, producing derived features through standardized interfaces.

The Digital Brain operates as the top-level service coordinating these base services and managing interaction with the User Profile. It functions as an orchestrator that schedules service execution, resolves dependency ordering, and unifies multimodal outputs into a coherent processing pipeline. It receives processed outputs, accesses measurements through synchronization mechanisms, and implements the PKG construction pipeline (Section 4). Through this pipeline, the Digital Brain transforms conversational interactions and multimodal observations into validated semantic relationships. It both reads from and writes to the User Profile, retrieving existing knowledge for reasoning and integrating newly validated triples into the PKG.

The User Profile extends the basic DT Profile to incorporate comprehensive user representations. Static characteristics, such as demographic attributes, preferences, and long-term goals, are stored in structured fields. The dynamic component is the Personal Knowledge Graph (PKG), a labeled directed graph $G_{PKG} = (V, E, L, P)$ where V contains entities and concepts extracted from user data, E represents semantic relations validated against ontological constraints, L encodes type labels derived from ontological alignment, and P associates temporal and contextual metadata with graph elements. The PKG evolves as validated relations produced by the Digital Brain are integrated, maintaining a temporally coherent knowledge structure that supports reasoning across sources and modalities.

3.2. Application Layer

The Application Layer mediates interactions between the internal architecture and external systems, managing bidirectional information flow while maintaining a clear separation from the DT Layer. It supports scalable operation across multiple concurrent DTs by standardizing data exchange patterns and ensuring that applications interact with the system through uniform, asynchronous interfaces. Let $\mathcal{A} = \{A_1, A_2, \ldots, A_k, \ldots, A_{|\mathcal{A}|}\}$ denote the set of applications active in the system, where each application A_k exchanges data with the DT Layer and external services.

Applications fall into two complementary categories. Native applications consume data produced by DTs, aggregating information across multiple replicas to deliver services such as activity visualization, health monitoring dashboards, or analytics over accumulated user data. Their outputs are exposed to external consumers through user interfaces or API endpoints. Connector applications establish synchronization channels with external service providers, such as calendar systems, email platforms, or third-party APIs, and import structured data streams into the internal architecture. This external information enriches the HDT's Digital Brain with contextual elements that extend beyond direct sensing capabilities, supporting integration into the PKG construction pipeline. Incoming streams are temporally aligned with measurements produced by DTs, ensuring coherent multimodal integration within the HDT.

Communication between applications and internal components is handled through the Model Context Protocol (MCP), detailed in the following subsection. Applications expose their outputs as structured observations that the Digital Brain can query alongside data produced by the DT Layer. MCP standardizes these exchanges, enabling loose coupling and interoperability across heterogeneous sources and consumers.

3.2.1. Model Context Protocol

The MCP server provides the communication infrastructure that manages information flow between data producers in the DT Layer and consumers throughout the system. DTs expose their measurements and processed outputs through the protocol, while applications and internal services access this information through standardized query interfaces. This decoupling enables flexible data access patterns without requiring direct coupling between components.

Data sources in the system exhibit heterogeneous characteristics. Digital replicas produce continuous time-stamped measurement streams; applications generate discrete events triggered by user interactions or scheduled operations; and textual replicas introduce free-form linguistic content at irregular intervals. These channels differ in temporal granularity and structural encoding, requiring consumers, particularly the HDT's Digital Brain, to integrate sensor readings, application events, and conversational inputs within a coherent multimodal representation.

The MCP server addresses these challenges through enrichment and standardized access. It receives raw measurements $m_{e_i}^j = (a_{e_i}^j, v_{e_i}^j, \tau_{e_i}^j)$ and augments them with metadata such as source identifiers, quality indicators, and device state at acquisition time. Timestamps are normalized within a global temporal reference frame to allow alignment across asynchronous sensing modalities. The enriched observation is denoted by $\tilde{m}_{e_i}^j$, and the server maintains a persistent repository indexing observations by entity and temporal position.

Data access follows a three-phase interaction model, i.e. negotiation, discovery, and retrieval. Negotiation establishes the communication channel and operational constraints; discovery enumerates source entities and their metadata; and retrieval provides structured access through three query patterns inspired by time-series management in IoT systems [50].

Temporal window queries extract observations within a specified interval:

$$Q_T(e_i, t_{\text{start}}, t_{\text{end}}) = \{\tilde{m}_{e_i}^j \mid \tau_{e_i}^j \in [t_{\text{start}}, t_{\text{end}}]\} \tag{2}$$

Conditional queries filter observations that satisfy given value- or metadata-based predicates:

$$Q_C(e_i, \text{conditions}) = \{\tilde{m}_{e_i}^j \mid \tilde{m}_{e_i}^j \text{ satisfies conditions}\} \tag{3}$$

where conditions specify thresholds or categorical constraints, supporting event detection and anomaly identification without retrieving full temporal sequences. Statistical queries compute aggregated metrics over temporal windows:

$$Q_S(e_i, t_{\text{start}}, t_{\text{end}}, f) = f(\{\tilde{m}_{e_i}^j \mid \tilde{m}_{e_i}^j \in Q_T(e_i, t_{\text{start}}, t_{\text{end}})\}) \tag{4}$$

providing mean, maximum, minimum, or variance through function $f \in \{\text{mean}, \max, \min, \text{var}\}$ for trend analysis and pattern recognition.

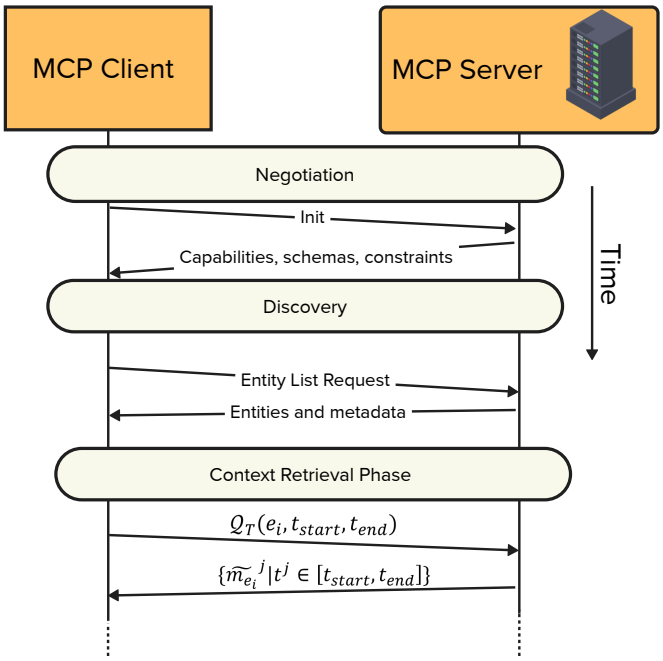


Figure 2: MCP Protocol Interaction Phases

These operations allow the Digital Brain to retrieve measurements around conversational timestamps for contextual grounding, select observations relevant to reasoning tasks, and compute statistical summaries for pattern recognition. Contextual observations retrieved through MCP are treated as optional enrichment signals. If certain sources are unavailable due to sensor dropouts or connectivity issues, the pipeline continues operating using the available observations, potentially producing a less enriched but still valid knowledge representation. MCP also supports confidence-scored observations and basic validation mechanisms, enabling robust integration of multimodal inputs into the PKG construction pipeline. Figure 2 illustrates the interaction phases between the Digital Brain and the repository.

4. Digital Brain Pipeline

The Digital Brain service implements a multi-stage pipeline that transforms conversational interactions and contextual observations into validated semantic relationships stored in the PKG. The pipeline progressively enriches linguistic content with structured multimodal data, aligns heterogenous inputs in a unified temporal and semantic space, and validates extracted knowledge against formal ontological constraints. Validated relationships are then organized within a hierarchical navigation framework that supports scalable graph traversal and reasoning. Figure 3 presents the overall pipeline architecture, while Table 2 summarizes the mathematical notation used throughout this work.

4.1. Triple Acquisition

The Digital Brain implements a three-stage process that acquires semantic triples from conversational interactions and

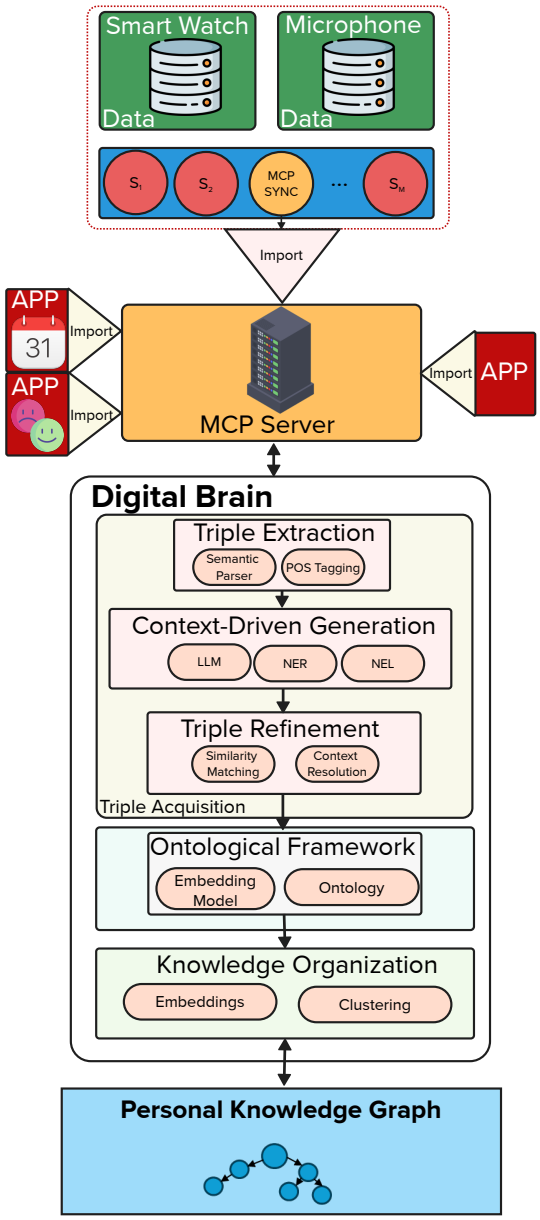


Figure 3: Digital Brain Pipeline

contextual observations, progressively transforming unstructured linguistic content into contextually grounded knowledge. The pipeline performs i) triple extraction from natural language transcriptions, ii) context-driven triple generation through temporal retrieval windows, and iii) triple refinement to resolve redundancies and inconsistencies prior to alignment.

Triple Extraction: Conversational inputs introduce unstructured linguistic content that carries semantic information about user activities, intentions, and subjective states. Unlike the structured observations produced by DTs and applications, natural language conveys meaning through flexible linguistic expressions that require interpretation to extract formal knowledge representations. Within the HDT architecture, a digital replica associated with an audio capture device may process

Table 2: Notation Summary for PKG Creation Pipeline

Symbol	Description
$\mathcal{W}(t_u)$	Temporal window: $[t_u - \delta^-, t_u + \delta^+]$
δ^-, δ^+	Temporal window bounds (past and future)
$\delta - mode$	Context depth parameter [0, 1]
$C(t_u)$	Context: observations within temporal window
$\mathcal{E}_1$	Triple extraction function
$\mathcal{E}_2$	Triple generation function
$\mathcal{E}_3$	Triple refinement function
$tr_i^{t_u}$	Triple with index i and timestamp t_u
$\mathcal{T}_{raw}$	Set of raw triples before refinement
$\mathcal{T}_R$	Set of refined triples
tr_i^R	Refined triple with index i
$\mathcal{O}$	Ontology specification
$\mathcal{K}$	Set of semantic classes
$\mathcal{P}$	Set of standardized properties
dom	Domain function: $\mathcal{P} \to \mathcal{K}$
ran	Range function: $\mathcal{P} \to \mathcal{K}$
(s_i, p_i, o_i)	Triple components
$\mathcal{L}$	Set of textual labels
ϕ	Embedding function: $\mathcal{L} \to \mathbb{R}^d$
$\mathbf{e}_p, \mathbf{e}_c$	Pre-computed embeddings
$\mathbf{v}_s, \mathbf{v}_p, \mathbf{v}_o$	Component embeddings
τ	Similarity threshold
$\alpha_1, \alpha_2, \alpha_3$	Alignment scores
$\mathcal{T}_b$	Set of broader topics
$\mathcal{T}_n$	Set of narrower topics

input through Automatic Speech Recognition (ASR), producing textual transcriptions that are made available to the Digital Brain service.

The triple extraction stage transforms these transcriptions into structured semantic representations suitable for PKG integration. Given a transcription $\mathcal{T}_{t_u}$ produced at time t_u, the Digital Brain applies an extraction function:

$$\mathcal{E}_1 : \mathcal{T}_{t_u} \to \{tr_1^{t_u}, tr_2^{t_u}, \ldots, tr_n^{t_u}\} \tag{5}$$

which identifies subject–predicate–object relationships, where $n \geq 0$ denotes on the informational density of the utterance. Each triple is represented as:

$$tr_i^{t_u} = (s_i, p_i, o_i)^{t_u} \tag{6}$$

where s_i denotes the subject entity, p_i the predicate relationship, o_i the object entity, and t_u marks the temporal moment of utterance. The extraction function $\mathcal{E}_1$ is performed by LLM capable of segmenting the linguistic structure and identifying embedded relationships, producing formal representations suitable for downstream PKG integration.

Context-driven triple generation: Linguistic triples capture the content of an utterance but often lack contextual grounding. Many statements rely on implicit references, e.g., fatigue, exertion, or location, that are best interpreted alongside concurrent physiological or environmental measurements.

To retrieve such information, the Digital Brain defines a temporal window around each utterance:

$$\mathcal{W}(t_u) = [t_u - \delta^-, t_u + \delta^+] \tag{7}$$

where δ^- captures prior context and δ^+ accounts for asynchronous pipeline execution, enabling incorporation of observations slightly after the utterance but before processing. This window captures observations that are temporally proximate to the conversational event while maintaining computational efficiency in MCP queries. The Digital Brain retrieves observations from available entities by invoking temporal window queries on the MCP server over all entities E:

$$C(t_u) = \bigcup_{e_k \in E} Q_T(e_k, t_u - \delta^-, t_u + \delta^+) \tag{8}$$

These observations feed a generation function:

$$\mathcal{E}_2 : C(t_u) \to \{tr_j^{t_u} \mid j = 1, \ldots, m\} \tag{9}$$

producing triples that encode quantitative measurements or environmental conditions, where $m \geq 0$ denotes the number of triples generated from structured observations, varying according to the availability and relevance of contextual data within the temporal window. At this stage, the Digital Brain thus maintains two complementary triple sets, i.e. $\mathcal{E}_1$-derived linguistic triples and $\mathcal{E}_2$-derived contextual triples, yielding a total of $n+m$ triples requiring subsequent refinement.

Triple refinement: Combining linguistic and structured sources introduces redundancy and potential inconsistencies. Users may describe the same event using different expressions, such as stating "I went to the gym this morning" followed by "I had a workout session today", while sensor-derived information may duplicate linguistically extracted triples, such as GPS data confirming "I'm at the gym". All this explicit duplication that must be resolved before PKG integration.

The refinement function $\mathcal{E}_3$ addresses these issues by consolidating the combined triple sets from both extraction and generation stages. Let $\mathcal{T}_{raw} = \{tr_1^{t_u}, \ldots, tr_{n+m}^{t_u}\}$ denote the union of extracted and generated triples. The refinement function processes this set alongside the contextual observations:

$$\mathcal{E}_3 : (\mathcal{T}_{raw}, C(t_u)) \to \mathcal{T}_R \tag{10}$$

where $\mathcal{T}_R = \{tr_1^R, \ldots, tr_k^R\}$ denotes the set of refined triples with $k \leq n + m$ elements after deduplication and validation. Redundant or semantically equivalent triples are merged, inconsistent or low-confidence relationships are removed, and ambiguous cases are resolved by leveraging contextual observations $C(t_u)$. The resulting refined set preserves semantic coherence, eliminates duplication, and integrates linguistic and sensor-derived information into a unified representation suitable for ontological alignment.

4.2. Ontological Framework

The refined triple set $\mathcal{T}_R = \{tr_1^R, \ldots, tr_k^R\}$ may still contain semantic inconsistencies or use terms that diverge from the ontology's standardized vocabulary. These deviations can violate domain constraints or introduce ambiguities that compromise interoperability and the semantic integrity of the PKG. For notational simplicity, each refined triple is written as $tr_i^R = (s_i, p_i, o_i)$, with the utterance timestamp omitted.

To ensure semantic correctness, the Digital Brain applies an ontology-based alignment mechanism that validates and standardizes each triple before PKG integration. The ontology is defined as $\mathcal{O} = (\mathcal{K}, \mathcal{P}, \text{dom}, \text{ran})$, where $\mathcal{K}$ is the set of semantic classes, $\mathcal{P}$ the set of standardized properties, and dom, ran map each property to the valid subject and object classes. These mappings enforce type constraints that each aligned triple must satisfy.

To map extracted terms to their ontological counterparts, the system uses embedding-based semantic similarity. Let $\mathcal{L} = \mathcal{K} \cup \mathcal{P} \cup \{s_i, p_i, o_i \mid tr_i^R \in \mathcal{T}_R\}$ denote the set of labels considered during alignment. An embedding function $\phi : \mathcal{L} \to \mathbb{R}^d$ projects these labels into a d-dimensional vector space where semantically related elements lie in close proximity.

For computational efficiency, embeddings for all classes and properties are pre-computed: $\mathbf{e}_p \in \mathbb{R}^d$ for each property $p \in \mathcal{P}$ and $\mathbf{e}_c \in \mathbb{R}^d$ for each class $c \in \mathcal{K}$. The triple components are instead encoded at runtime as $\mathbf{v}_s = \phi(s_i)$, $\mathbf{v}_p = \phi(p_i)$, and $\mathbf{v}_o = \phi(o_i)$. Similarity is computed using cosine similarity:

$$\text{sim}(\mathbf{v}, \mathbf{e}) = \frac{\mathbf{v} \cdot \mathbf{e}}{\|\mathbf{v}\|_2 \cdot \|\mathbf{e}\|_2} \tag{11}$$

where $\mathbf{v}$ represents an extracted term embedding (e.g., $\mathbf{v}_s$, $\mathbf{v}_p$, or $\mathbf{v}_o$) and $\mathbf{e}$ represents an ontological element embedding (e.g., $\mathbf{e}_c$ or $\mathbf{e}_p$), with $\cdot$ denoting the dot product and $\|\cdot\|_2$ the Euclidean norm.

A potential limitation of embedding-based alignment concerns vocabulary mismatch, where a user expression is semantically related to an ontology class or property but appears distant in the embedding space due to slang, paraphrasing, or domain-specific jargon. For instance, informal expressions such as "working out" may not be lexically identical to the ontological class "ExerciseActivity", potentially reducing cosine similarity despite conceptual equivalence. The proposed alignment mechanism mitigates this risk through multiple robustness layers. First, the use of three parallel matching strategies reduces dependence on any single triple component: even if one element suffers from lexical distortion, the remaining components can guide the alignment through ontological constraints. Second, domain and range restrictions limit candidate classes and properties to structurally compatible subsets, preventing spurious matches driven solely by vector proximity. Third, the acceptance threshold τ filters low-confidence alignments, ensuring that poorly grounded mappings are excluded from the PKG while remaining available in the digital replica for future reconsideration.

Finally, contextual enrichment introduced during the refinement stage (Section 4.1) reduces lexical variability by consolidating semantically equivalent expressions before ontological validation. Together, these mechanisms improve tolerance to linguistic variability without requiring domain-specific embedding retraining. The alignment process evaluates each triple along three parallel matching strategies, each initiating from a different component of the triple and propagating through ontological constraints. Running all strategies in parallel increases robustness: if one triple component is noisy or poorly extracted, the remaining strategies can still recover a valid alignment.

We detail the predicate-first strategy; the others follow analogous procedures. The process begins by selecting the most similar ontology property:

$$p_1^* = \arg\max_{p \in \mathcal{P}} \text{sim}(\mathbf{v}_p, \mathbf{e}_p) \tag{12}$$

Domain and range constraints are then retrieved, where the superscript (1) denotes quantities specific to the predicate-first strategy:

$$\mathcal{K}_{\text{dom}}^{(1)} = \text{dom}(p_1^*), \quad \mathcal{K}_{\text{ran}}^{(1)} = \text{ran}(p_1^*) \tag{13}$$

These constraints restrict the class search space for subject and object alignment:

$$c_{s,1}^* = \arg\max_{c \in \mathcal{K}_{\text{dom}}^{(1)}} \text{sim}(\mathbf{v}_s, \mathbf{e}_c), \quad c_{o,1}^* = \arg\max_{c \in \mathcal{K}_{\text{ran}}^{(1)}} \text{sim}(\mathbf{v}_o, \mathbf{e}_c) \tag{14}$$

The predicate-first alignment score aggregates similarity across all components:

$$\alpha_1 = \frac{1}{3}\left(\text{sim}(\mathbf{v}_p, \mathbf{e}_{p_1^*}) + \text{sim}(\mathbf{v}_s, \mathbf{e}_{c_{s,1}^*}) + \text{sim}(\mathbf{v}_o, \mathbf{e}_{c_{o,1}^*})\right) \tag{15}$$

The subject-first strategy proceeds analogously by first identifying the subject class $c_{s,2}^* = \arg\max_{c \in \mathcal{K}} \text{sim}(\mathbf{v}_s, \mathbf{e}_c)$, then constraining property search to $\mathcal{P}_{\text{subj}}^{(2)} = \{p \in \mathcal{P} : \text{dom}(p) = c_{s,2}^*\}$, and finally aligning the object within $\text{ran}(p_2^*)$. Similarly, the object-first strategy initiates from object alignment, constrains properties by range, and concludes with subject matching. Both strategies compute their respective scores α_2 and α_3 through averaging component similarities.

The final alignment score selects the strongest path:

$$\text{score}(tr_i^R) = \max(\alpha_1, \alpha_2, \alpha_3) \tag{16}$$

A triple is accepted if $\text{score}(tr_i^R) \geq \tau$, where $\tau \in [0, 1]$ determines valid alignments. Triple below the threshold are excluded from the PKG but retained in the HDT's digital replica to preserve a complete historical record while ensuring the semantic consistency of the inferred knowledge.

Algorithm 1 summarizes the full alignment procedure, detailing the three matching strategies and their computational implementation. To improve clarity, we provide a step-by-step explanation of the algorithm and its correspondence with Equations (12) to (16). Line 1 initializes the process by computing the embeddings of the triple components, i.e., $v_s = \phi(s_i)$, $v_p = \phi(p_i)$, and $v_o = \phi(o_i)$. These vectors are used throughout the three matching strategies. Lines 2–7 implement the predicate-first strategy. Line 3 corresponds to Equation (12), selecting the ontology property p_1^* that maximizes cosine similarity with the extracted predicate. Line 4 retrieves the domain and range constraints of p_1^* as defined in Equation (13). Lines 5 and 6 perform constrained class matching for subject and object, corresponding to Equation (14), restricting the search space to $K_{\text{dom}}^{(1)}$ and $K_{\text{ran}}^{(1)}$. Line 7 computes the aggregated alignment score α_1 as defined in Equation (15). Lines 8–14 describe the subject-first strategy. Line 9 identifies the most similar subject class

over the full ontology class set. Line 10 restricts the candidate property set to those whose domain matches the selected subject class. Lines 11–13 perform predicate and object alignment under these constraints, and Line 14 computes the corresponding score α_2 through average similarity aggregation. Lines 15–21 implement the object-first strategy symmetrically. Line 16 identifies the best matching object class, Line 17 restricts candidate properties by range compatibility, Lines 18–20 complete predicate and subject alignment, and Line 21 computes α_3. Finally, Lines 22–24 select the maximum score among the three strategies, as defined in Equation (16), and return the final alignment score for validation against the threshold τ.

Algorithm 1 Ontology-based Semantic Alignment

Require: Refined triple $tr_i^R = (s_i, p_i, o_i)$, ontology $O = (\mathcal{K}, \mathcal{P}, \text{dom}, \text{ran})$, embedding function ϕ, threshold τ
Ensure: Alignment score $\text{score}(tr_i^R)$

1: Compute embeddings: $\mathbf{v}_s \leftarrow \phi(s_i)$, $\mathbf{v}_p \leftarrow \phi(p_i)$, $\mathbf{v}_o \leftarrow \phi(o_i)$
2: **Predicate-first Strategy**
3: $p_1^* \leftarrow \arg\max_{p \in \mathcal{P}} \text{sim}(\mathbf{v}_p, \mathbf{e}_p)$
4: $\mathcal{K}_{\text{dom}}^{(1)} \leftarrow \text{dom}(p_1^*)$, $\mathcal{K}_{\text{ran}}^{(1)} \leftarrow \text{ran}(p_1^*)$
5: $c_{s,1}^* \leftarrow \arg\max_{c \in \mathcal{K}_{\text{dom}}^{(1)}} \text{sim}(\mathbf{v}_s, \mathbf{e}_c)$
6: $c_{o,1}^* \leftarrow \arg\max_{c \in \mathcal{K}_{\text{ran}}^{(1)}} \text{sim}(\mathbf{v}_o, \mathbf{e}_c)$
7: $\alpha_1 \leftarrow \frac{1}{3}\left(\text{sim}(\mathbf{v}_p, \mathbf{e}_{p_1^*}) + \text{sim}(\mathbf{v}_s, \mathbf{e}_{c_{s,1}^*}) + \text{sim}(\mathbf{v}_o, \mathbf{e}_{c_{o,1}^*})\right)$
8: **Subject-first Strategy**
9: $c_{s,2}^* \leftarrow \arg\max_{c \in \mathcal{K}} \text{sim}(\mathbf{v}_s, \mathbf{e}_c)$
10: $\mathcal{P}_{\text{subj}}^{(2)} \leftarrow \{p \in \mathcal{P} : \text{dom}(p) = c_{s,2}^*\}$
11: $p_2^* \leftarrow \arg\max_{p \in \mathcal{P}_{\text{subj}}^{(2)}} \text{sim}(\mathbf{v}_p, \mathbf{e}_p)$
12: $\mathcal{K}_{\text{ran}}^{(2)} \leftarrow \text{ran}(p_2^*)$
13: $c_{o,2}^* \leftarrow \arg\max_{c \in \mathcal{K}_{\text{ran}}^{(2)}} \text{sim}(\mathbf{v}_o, \mathbf{e}_c)$
14: $\alpha_2 \leftarrow \frac{1}{3}\left(\text{sim}(\mathbf{v}_s, \mathbf{e}_{c_{s,2}^*}) + \text{sim}(\mathbf{v}_p, \mathbf{e}_{p_2^*}) + \text{sim}(\mathbf{v}_o, \mathbf{e}_{c_{o,2}^*})\right)$
15: **Object-first Strategy**
16: $c_{o,3}^* \leftarrow \arg\max_{c \in \mathcal{K}} \text{sim}(\mathbf{v}_o, \mathbf{e}_c)$
17: $\mathcal{P}_{\text{obj}}^{(3)} \leftarrow \{p \in \mathcal{P} : \text{ran}(p) = c_{o,3}^*\}$
18: $p_3^* \leftarrow \arg\max_{p \in \mathcal{P}_{\text{obj}}^{(3)}} \text{sim}(\mathbf{v}_p, \mathbf{e}_p)$
19: $\mathcal{K}_{\text{dom}}^{(3)} \leftarrow \text{dom}(p_3^*)$
20: $c_{s,3}^* \leftarrow \arg\max_{c \in \mathcal{K}_{\text{dom}}^{(3)}} \text{sim}(\mathbf{v}_s, \mathbf{e}_c)$
21: $\alpha_3 \leftarrow \frac{1}{3}\left(\text{sim}(\mathbf{v}_o, \mathbf{e}_{c_{o,3}^*}) + \text{sim}(\mathbf{v}_p, \mathbf{e}_{p_3^*}) + \text{sim}(\mathbf{v}_s, \mathbf{e}_{c_{s,3}^*})\right)$
22: **Select best strategy**
23: $\text{score}(tr_i^R) \leftarrow \max(\alpha_1, \alpha_2, \alpha_3)$
24: **return** $\text{score}(tr_i^R)$

4.3. PKG Construction

The validated triples produced by the ontological alignment stage constitute atomic knowledge units that must be organized into a coherent, navigable structure. Once a triple exceeds the similarity threshold τ, it is integrated into the PKG G_{PKG}, transforming isolated semantic relationships into a unified graph representation.

The PKG is structured into two complementary layers. The knowledge layer contains entities and their semantic relationships, forming the core graph content where validated triples materialize as typed nodes and edges. The navigation layer provides a topic-based hierarchy that organizes these relationships

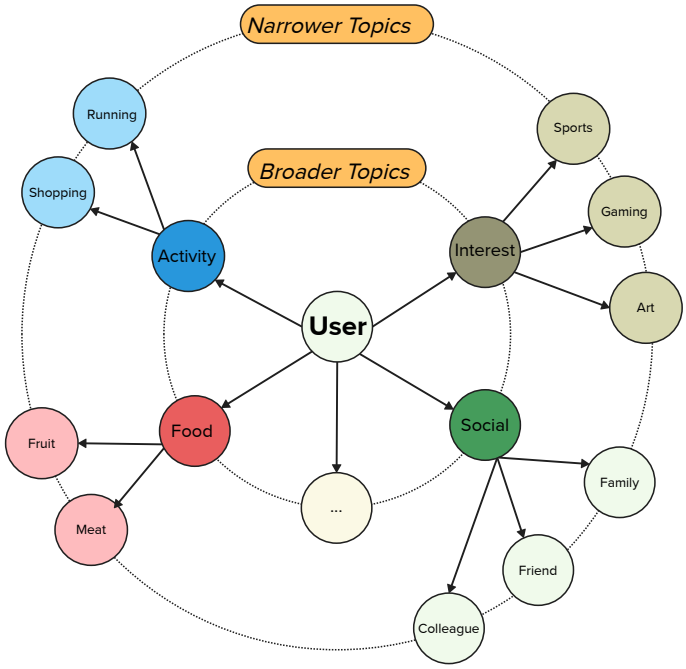


Figure 4: Hierarchical Topic Organization in PKG

into a logical structure, enabling efficient traversal and contextual filtering. Topics serve purely as organizational metadata and do not represent knowledge themselves; this separation allows multiple navigation perspectives over the same underlying semantic graph without altering its meaning.

To implement this hierarchical navigation, we adopt a Simple Knowledge Organization System (SKOS) structure, a W3C standard for representing knowledge organization structures. SKOS provides a formal vocabulary for expressing hierarchical relationships between concepts through "skos:broader" and "skos:narrower" properties, enabling the construction of taxonomies that span multiple levels of abstraction. Within our framework, we employ a two-level taxonomy. The first level includes broader topics $\mathcal{T}_b$, representing high-level domains such as Health, Finance, Social, Interests, and Activities. The second level consists of narrower topics $\mathcal{T}_n$, representing subcategories within each broader domain, such as Vital Signs under Health, Expenses under Finance, or Walking under Activities. Broader topics provide coarse organization, while narrower topics enable fine-grained filtering that improves query efficiency during graph updates and traversal.

Each validated triple tr_i^R is classified into a unique topic pair (T_b, T_n) with $T_b \in \mathcal{T}_b$ and $T_n \in \mathcal{T}_n$. Figure 4 illustrates the hierarchical structure showing how broader categories branch into narrower topics relevant to different aspects of the user's profile.

Let $C_{\text{topic}} : \mathcal{T}_R \rightarrow \mathcal{T}_b \times \mathcal{T}_n$ denote the topic classification function that maps each refined triple to a broader-narrower topic pair. The function uses embedding-based semantic similarity between triple components and topic labels to determine the best-fitting broader and narrower categories. Given a refined triple. Given a refined triple tr_i^R with its ontologi-

cal types (c_s^*, c_p^*, c_o^*), classification considers the semantic content expressed by the triple to find the most appropriate topic pair. The process employs semantic matching to identify existing topics that closely correspond to the triple's content, while introducing new topic pairs when semantic distance from all existing categories exceeds a coherence threshold.

The PKG construction follows a dynamic approach that balances topic reuse with taxonomy evolution. When processing a new triple with classified topics (T_b, T_n), the system reuses existing topics when semantic matches are found, creating new ones only when necessary. This limits unnecessary fragmentation while supporting expansion into new knowledge domains.

Formally, the PKG is represented as a labeled property graph $G_{PKG} = (V, E, L, P)$ where V is the set of entity nodes, E the set of directed edges representing semantic relationships, L the labeling function assigning ontological types to nodes and edges, and P the property function storing metadata, including topic annotations and temporal information, with graph elements. The PKG materializes validated triples as a property graph centered on the user, where entities become nodes labeled with their ontological types determined during the semantic alignment phase. A validated triple $tr_i^R = (s_i, p_i, o_i)$ with ontological types (c_s^*, c_p^*, c_o^*) produces two entity nodes and one typed relationship: the subject creates a node of type c_s^*, the object creates a node of type c_o^*, and the predicate forms a typed directed edge connecting these nodes. The topic classification (T_b, T_n) is stored as edge metadata rather than as additional nodes, preserving structural simplicity while enabling efficient topic-based navigation. Queries can thus retrieve subgraphs by selecting edges matching specific topic criteria, generating domain-specific views over the knowledge structure without duplicating nodes or altering the underlying topology.

5. Simulation Setup

5.1. Language Models and Semantic Infrastructure

The PKG creation pipeline described in Section 4 comprises sequential stages that progressively transform conversational content into structured knowledge. The extraction function $\mathcal{E}_1$ produces candidate triples from natural language; the contextualization function $\mathcal{E}_2$ enriches them with structured observations retrieved via the MCP; and the refinement function $\mathcal{E}_3$ consolidates redundancies and resolves semantic ambiguities. Following triple acquisition, ontological alignment validates each relationship against formal semantic constraints, and topic classification assigns validated triples to the hierarchical navigation framework. All these stages depend on language model capabilities instantiated through the configurations evaluated in Section 6.

We employed Groq as the inference provider for all language model operations to enable high-throughput token generation and support the efficient execution of the knowledge construction pipeline. The evaluated architectures span multiple parameter scales and design paradigms. Specifically, *gpt-oss-120b* and *gpt-oss-20b* as representative dense architectures offering different computational footprints, *llama-4-maverick-17b-128e-instruct* as a mid-range efficient configuration balancing inference speed with semantic understanding, *kimi-k2-instruct-0905* implementing mixture-of-experts topology with 32B active parameters to optimize computational allocation across specialized sub-networks, and *qwen3-32b* as a competitive 30B-class model demonstrating strong reasoning capabilities within moderate resource constraints. These models exhibit distinct trade-offs between inference cost and semantic reasoning ability. Smaller models provide faster execution at the potential cost of extraction precision, whereas larger models deliver richer semantic interpretation with higher compute requirements. In addition to model size, the operational cost of the pipeline also depends on the depth of contextual retrieval, since wider temporal windows may increase the amount of information processed by the triple acquisition stages. For this reason, in Section 6 we complement the accuracy evaluation with a token-consumption analysis across representative δ configurations, using token consumption as a stable proxy for practical inference cost. The δ context depth parameter directly controls the amount of contextual information retrieved around each utterance and therefore influences the token budget of the triple acquisition stages. The system supports two configurations, namely δ-mode 0 and δ-mode 1. The former represents a more economical setting, relying on a limited contextual window, while the latter leverages a broader context to achieve higher semantic accuracy at the expense of increased token consumption. We evaluate each model across the core pipeline components: semantic extraction $\mathcal{E}_1$, multimodal contextualization $\mathcal{E}_2$, triple refinement $\mathcal{E}_3$, and the topic classification C_{topic}.

For the ontological alignment stage, we rely on the Schema.org semantic framework (version 29.3, released September 2025), which includes over 800 classes and 1,450 properties. Domain and range constraints enforce type consistency: each property $p \in \mathcal{P}$ specifies valid subject types via $\text{dom}(p)$ and valid object types via $\text{ran}(p)$, supporting the three-path alignment mechanism described in Algorithm 1. Alignment uses embedding-based similarity between extracted terms and the Schema.org vocabulary. Embeddings are generated via Cohere's *embed-multilingual-v3.0*[1], implementing $\phi : \mathcal{L} \to \mathbb{R}^{1024}$, which projects textual labels into a 1024-dimensional semantic vector space. Prior to processing conversations, embeddings for all ontological elements are pre-computed, producing reference vectors $\mathbf{e}_p \in \mathbb{R}^{1024}$ for each property and $\mathbf{e}_c \in \mathbb{R}^{1024}$ for each class. Triple components are embedded at runtime and compared via cosine similarity to identify semantically equivalent ontological representations.

The knowledge organization stage leverages DBpedia[2] to support topic label resolution and hierarchical validation within the SKOS taxonomy. During topic classification, the language model performs entity linking to DBpedia resources corresponding to each validated triple (s, p, o). SPARQL queries over Wikipedia category hierarchies retrieve semantic neigh-

[1] https://cohere.com
[2] https://www.dbpedia.org

borhoods used to assign broader topics $T_b \in \mathcal{T}_b$ from upper-level categories, and narrower topics $T_n \in \mathcal{T}_n$ via more specific category memberships. SKOS relationships embedded in DBpedia's category system ensure consistency between broader and narrower topic pairs.

5.2. Pipeline Parameters

The pipeline relies on a set of operational parameters that configure each processing stage. The contextual retrieval window is defined as $\delta^- = 10$ minutes before the utterance and $\delta^+ = 10$ minutes after it, capturing temporally proximate observations while keeping MCP queries efficient. All language model operations, i.e. extraction, contextualization, refinement, and topic classification, use a temperature parameter of $T = 0.3$ and a maximum generation length of 2048 tokens, balancing determinism, semantic flexibility, and computational cost. The ontological alignment employs a similarity threshold of $\tau = 0.70$, representing the minimum score required for triple validation.

6. Results

The experimental evaluation examines the PKG creation pipeline across multiple dimensions, assessing the effectiveness of different language model configurations for triple acquisition, the contribution of the ontological alignment to semantic quality, and the overall coherence of the constructed knowledge graphs. The experiments combine conversational data from public extraction benchmarks with synthetic multimodal scenarios designed to test the end-to-end pipeline under controlled conditions. The following subsections present comparative performance analyses and ablation studies quantifying the impact of each pipeline component.

6.1. Language Model Performance in Triple Acquisition

The three-stage triple acquisition process depends strongly on language model capabilities, with each stage introducing different semantic and computational demands. We evaluated five models of varying size and architecture across extraction ($\mathcal{E}_1$), contextualization ($\mathcal{E}_2$), and refinement ($\mathcal{E}_3$) to determine optimal configurations for each stage. The evaluation encompasses *gpt-oss-120b* and *gpt-oss-20b* as representative dense architectures, *llama-4-maverick-17b-128e-instruct* as a mid-range efficient configuration, *kimi-k2-instruct-0905* implementing mixture-of-experts topology with 32B active parameters, and *qwen3-32b* as a competitive 30B-class model. All models support structured JSON output, ensuring consistent triple formatting without post-processing.

For the extraction stage $\mathcal{E}_1$, we used the manually annotated benchmark from *Tiny But Mighty* [51] dataset. It distinguishes between simple sentences requiring straightforward subject-predicate-object parsing and complex sentences exhibiting nested clauses, coordination structures, or implicit semantic relationships that demand deeper linguistic reasoning. The dataset contains 2,276 triples from simple sentences and 719 from syntactically complex ones. We sampled 20% from each category, yielding 455 simple and 144 complex cases. This ensures coverage of both straightforward structures and utterances involving nested clauses, implicit relations, or coordination phenomena.

Table 3: Extraction Performance ($\mathcal{E}_1$) on Simple Sentences (455 test triples). Thresholds: Cosine $\tau_c = 0.7$, Jaccard $\tau_j = 0.5$, Fuzzy $\tau_f = 0.6$. Avg: Average Similarity.

Model	Cosine		Jaccard		Fuzzy	
	F1	Avg	F1	Avg	F1	Avg
llama-4-maverick-17b	.81	.74	.22	.48	.71	.65
gpt-oss-20b	.62	.71	.15	.38	.58	.63
kimi-k2-instruct (32B)	.78	.73	.19	.44	.68	.66
qwen3-32b	.76	.72	.18	.43	.66	.64
gpt-oss-120b	.85	.77	.26	.50	.75	.68

Table 4: Extraction Performance ($\mathcal{E}_1$) on Complex Sentences (144 test triples). Thresholds: Cosine $\tau_c = 0.7$, Jaccard $\tau_j = 0.5$, Fuzzy $\tau_f = 0.6$. Avg: Average Similarity.

Model	Cosine		Jaccard		Fuzzy	
	F1	Avg	F1	Avg	F1	Avg
llama-4-maverick-17b	.79	.72	.12	.41	.65	.58
gpt-oss-20b	.52	.60	.09	.32	.63	.62
kimi-k2-instruct (32B)	.77	.71	.16	.42	.67	.70
qwen3-32b	.75	.70	.15	.40	.63	.62
gpt-oss-120b	.81	.77	.28	.45	.79	.67

Each model generated candidate triples that were evaluated against ground truth using exact matching and embedding-based similarity measures. The evaluation captures both structural accuracy through exact matching criteria and semantic similarity through embedding-based measures, providing complementary perspectives on extraction quality. We report performance separately for simple and complex sentences to capture sensitivity to syntactic difficulty and identify architectural patterns that influence robustness.

Table 3 reports the results for simple sentences, which mainly require direct subject–predicate–object identification. The results reveal how models of different scales handle straightforward triple extraction tasks that constitute the majority of conversational content.

The complex sentence evaluation in Table 4 evaluates model performance on syntactically challenging utterances where semantic relationships may be implied through context or expressed through non-standard grammatical constructions.

The results show a clear correlation between model scale and extraction quality. *gpt-oss-120b* achieves the highest F1 score on both simple (0.85) and complex (0.81) inputs under cosine similarity, but *llama-4-maverick-17b-128e-instruct* remains competitive with only a modest reduction (4–5%) despite requiring substantially fewer parameters. Mid-range models (*kimi-k2-instruct* and *qwen3-32b*) achieve similar performance on simple sentences (F1: 0.75 and 0.78 respectively), while *gpt-oss-20b* performs markedly worse, especially on complex structures (F1: 0.52). Performance degradation from simple to complex inputs is limited for larger models (e.g., 2 points for *llama-*

4-maverick-17b-128e-instruct) but substantial for smaller ones (10 points for *gpt-oss-20b*). Cosine similarity systematically outperforms Jaccard by 50-60 points across all models, confirming that semantic level evaluation is more appropriate than lexical overlap for conversational data. Given these results, we adopt *llama-4-maverick-17b-128e-instruct* for the extraction stage $\mathcal{E}_1$, as its accuracy–latency trade-off provides the best balance for real-time operation.

6.2. End-to-End Pipeline Evaluation

While extraction performance determines the quality of initial candidates, the pipeline's effectiveness emerges from interactions among contextualization ($\mathcal{E}_2$), refinement ($\mathcal{E}_3$), and ontological alignment. Evaluating these components individually risks obscuring dependencies: contextualization provides temporal grounding that refinement must preserve, while ontological alignment removes noise introduced upstream. To assess the pipeline holistically, we apply an ablation methodology that measures how disabling specific components affects downstream reasoning. Rather than annotating intermediate outputs, we evaluate PKG utility through its ability to support inferential queries about user behavior, activities, and cross-modal relationships, reflecting the PKG's operational role in HDT architectures.

Synthetic evaluation data were generated for four persona types, each producing three days of multimodal observations including conversational utterances, calendar events, GPS traces, and physiological signals. Temporal patterns follow distributions from the American Time Use Survey [52], with sleep averaging 8 hours, work 6–8 hours, free time 3–4 hours, and personal care around 2 hours daily. Daily routines adhere to common constraints (morning activities 6–8am, work 9am–6pm, evening activities after 6pm). Personas differ in activity profiles to stress-test the pipeline across diverse reasoning scenarios.

From these data we construct 48 evaluation queries spanning five categories aligned with knowledge graph quality dimensions [53]: temporal grounding, cross-modal reasoning, deduplication, hallucination resistance, and completeness. Following [54], we adopt an extrinsic evaluation approach that measures how well the PKG supports downstream inference. Temporal queries assess ordering and alignment across sources; cross-modal queries test integration of linguistic and sensor evidence; deduplication queries measure consolidation of redundant facts; hallucination queries test resistance to non-existent events; completeness queries verify coverage. Each ablation configuration selectively disables pipeline components to quantify their contribution to reasoning accuracy.

Ground truth was labeled independently by three expert annotators with backgrounds in knowledge representation and human-computer interaction. Queries use formats enabling automatic scoring. Binary queries require yes/no answers for verifying event occurrence or temporal relationships. Exact match queries demand precise responses such as timestamps, counts, or entity names that can be evaluated through string comparison. Multiple choice queries present four options with a single correct answer, eliminating ambiguity in semantic interpretation. The evaluation measures PKG utility through query answering rather than structural graph metrics. For each configuration, the pipeline constructs a PKG from the multimodal data following Section 4. The resulting graph is serialized as a table where each row encodes a validated triple (s, p, o) with timestamp t_u and topic pair (T_b, T_n), preserving the labelled property graph structure $G_{PKG} = (V, E, L, P)$. Query answering uses *gpt-oss-120b*, which receives both the serialized PKG and the query. The model outputs strictly formatted responses (binary, exact string, or multiple choice), enabling direct comparison with ground truth. Accuracy is computed as the proportion of exact matches; false positive rate captures the likelihood of affirming non-existent events. We evaluated each configuration's capacity to support inferential reasoning about accumulated user knowledge, reflecting the PKG's operational purpose within HDT architectures where downstream reasoning accuracy determines system value more directly than intermediate structural properties.

The full configuration achieves 85% accuracy, serving as the baseline. Removing contextualization produces the largest drop: No-$\mathcal{E}_2$ falls to 52%, with severe losses in temporal grounding (30%) and cross-modal reasoning (45%), demonstrating that multimodal context integration constitutes an essential rather than incremental enhancement. Without access to sensor and application data around the utterance, the PKG cannot situate linguistic content in its physical or digital context. The $\mathcal{E}_1$-only configuration confirms this dependency, achieving just 35%, while $\mathcal{E}_1 + \mathcal{E}_2$ improves to 72%, indicating that contextualization provides most of the performance gain beyond extraction.

Refinement has a subtler effect. Disabling $\mathcal{E}_3$ increases the triple count by 38% (620 versus 450) and maintains good temporal and cross-modal accuracy (88% and 82% respectively), but sharply reduces deduplication performance (45%). This pattern reveals that refinement's primary contribution lies in consolidating redundant information rather than improving extraction accuracy directly. Additionally, the false positive rate doubles (12%), which indicates that refinement mainly contributes by consolidating redundant or inconsistent triples. Excess triples do not immediately corrupt reasoning but introduce noise that reduces answer coherence when multiple sources must be reconciled.

Ontological alignment primarily improves precision rather than recall. Without it (labelled as No-Onto), the system accepts 29% more triples (580 versus 450) but drops to 68% accuracy with the highest false positive rate (18%). Hallucination resistance decreases substantially (70% compared to 95%), showing that many unvalidated triples correspond to spurious or weakly grounded relationships. The ontology-free PKG is larger but less reliable, reflecting the fundamental need for semantic constraints to filter extraction artifacts, producing a knowledge base that is quantitatively larger but qualitatively weaker for supporting inferential reasoning. This trade-off highlights that large amounts of extracted information do not guarantee better reasoning unless validated against formal ontological structures.

While Groq enables high-throughput inference, throughput alone does not capture the practical deployment cost of the

Table 5: Query Answering Performance Across Pipeline Configurations (48 queries, 6 configurations). Category Performance represents accuracy on query subsets: T=Temporal grounding, C=Cross-modal reasoning, D=Deduplication, H=Hallucination resistance (higher = fewer false affirmations), F=Completeness. FPR: false positive rate across all queries.

Config	Accuracy	FPR	Triples	Category Performance (%)				
				T	C	D	H	F
Full	85%	5%	450	90	85	80	95	82
$\mathcal{E}_1$+$\mathcal{E}_2$	72%	12%	580	85	78	45	82	75
No-$\mathcal{E}_2$	52%	8%	180	30	45	75	92	55
No-$\mathcal{E}_3$	78%	12%	620	88	82	45	85	78
No-Onto	68%	18%	580	85	75	60	70	72
$\mathcal{E}_1$-only	35%	25%	120	25	20	30	60	28

pipeline. To isolate the effect of the context-depth parameter, we conducted an additional experiment on a reduced conversational dataset consisting of approximately 1,500 dialogue turns, corresponding to roughly 60 minutes of conversation. This experiment is intended to evaluate the operational impact of the context depth parameter. We compare two representative operating points of the pipeline, denoted as δ-mode 0 and δ-mode 1. Table 6 reports the resulting reasoning quality together with the total token consumption of the pipeline.

Table 6: Impact of the context depth parameter δ on reasoning quality and token consumption.

Config	Accuracy	FPR	Completeness	Tokens
δ-mode 0	66.25%	17.50%	80.00%	8,550
δ-mode 1	73.13%	15.00%	86.67%	25,076

The results show that increasing the context window raises the total token budget from 8,550 to 25,076 tokens. This increase is accompanied by a clear improvement in reasoning quality, with Accuracy increasing from 66.25% to 73.13% and Completeness improving from 80.00% to 86.67%, while the False Positive Rate decreases from 17.50% to 15.00%, indicating a lower number of false affirmations in the extracted knowledge.

Importantly, the number of extracted triples remains nearly unchanged across the two configurations, indicating that the wider context primarily improves the semantic quality and completeness of the extracted knowledge rather than increasing the size of the resulting graph.

Even in the configuration with the largest context window, the total token consumption remains well below the maximum context window of the adopted model (512,000 tokens for llama-4-maverick-17b-128e-instruct), demonstrating that the proposed pipeline operates comfortably within the model's operational limits.

Therefore, δ acts as a tunable cost-accuracy parameter. Smaller context windows provide a more economical configuration with lower token usage, whereas wider context windows enable more accurate semantic extraction. In practical deployments, this trade-off could translate into different service tiers, ranging from cost-efficient configurations to higher-accuracy settings with richer context reasoning.

6.3. Threshold Sensitivity Analysis

The ontological alignment threshold τ controls validation strictness and therefore directly influences PKG quality. To assess robustness, we evaluate performance for $\tau \in \{0.50, 0.60, 0.70, 0.80, 0.90\}$.

Figure 5 shows that the Full configuration achieves its best accuracy at $\tau = 0.70$ (85%), with degradation at both extremes. Permissive thresholds ($\tau \leq 0.60$) admit noisy triples that introduce spurious relationships, reducing accuracy to 72-82%. Conversely, strict thresholds ($\tau \geq 0.80$) reject valid relationships and produce sparse PKGs that lack sufficient information for reasoning, yielding 70% accuracy at $\tau = 0.90$ despite lower false positives. As expected, No-Onto remains constant at 68% since the threshold has no effect without ontological validation.

A threshold-driven crossover emerges between No-$\mathcal{E}_3$ and No-Onto, illustrating the quality-coverage trade-off in knowledge validation. At $\tau = 0.70$, No-$\mathcal{E}_3$ outperforms No-Onto (78% vs. 68%), showing that ontological validation compensates for the absence of refinement by filtering inconsistent triples. At $\tau = 0.90$, the trend reverses (No-$\mathcal{E}_3$: 62%, No-Onto: 68%), because strict thresholds over-filter unrefined triples, whose lexical variability leads to weaker matches. The Full configuration avoids this crossover because refinement standardizes terminology prior to validation.

Threshold sensitivity varies across configurations employing ontological validation. No-$\mathcal{E}_2$ shows limited accuracy (47-52%) due to the smaller triple set produced without contextualization, reducing the effect of threshold-based filtering. The $\mathcal{E}_1$-only baseline exhibits the strongest sensitivity, as its minimal input amplifies the impact of acceptance and rejection decisions. The selected threshold $\tau = 0.70$ provides robust validation without requiring precise calibration, exhibiting stable performance within a ± 0.10 range that accommodates embedding model variations or ontological vocabulary differences across different Schema.org versions or alternative semantic frameworks.

7. Deployment and Real-World Considerations

In addition, deployment considerations influence system behavior in realistic environments. The Digital Brain operates primarily as an asynchronous background service, allowing moderate processing delays for PKG construction without degrading user interaction quality. This design choice reduces strict

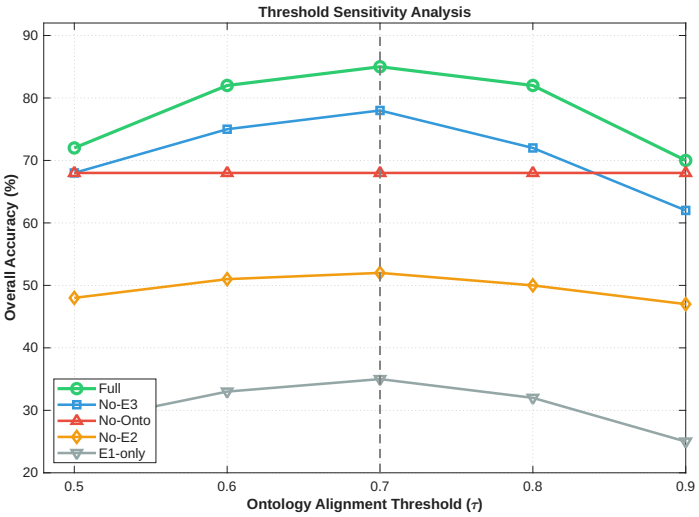


Figure 5: Threshold sensitivity analysis across pipeline configurations.

latency constraints and permits flexible cloud-based deployment, while edge-based execution and computation offloading remain possible extensions for privacy-sensitive or resource-constrained scenarios. In this context, the deployment and orchestration of resource-intensive components remain an important aspect for future system implementations. Mechanisms such as computation offloading and edge–cloud coordination have been widely studied to optimize the execution of delay-sensitive or resource-demanding applications in distributed environments. While the proposed architecture does not require ultra-low latency processing for the majority of its knowledge construction tasks, such mechanisms may become relevant when HDT services are deployed across heterogeneous infrastructures or when knowledge representations are reused by latency-sensitive services, such as conversational agents relying on the PKG. Recent works on mobile computation offloading and collaborative task execution [55] as well as dynamic HDT placement strategies at the network edge [56] provide promising directions for optimizing the deployment of knowledge-driven HDT services. These approaches could complement the proposed architecture by enabling adaptive placement of computational components across cloud–edge infrastructures while balancing reasoning accuracy, resource availability, and response latency.

An additional architectural consideration concerns the deployment strategy of the Digital Brain service, which performs LLM inference and ontology alignment. Unlike delay-sensitive conversational agents requiring sub-second reaction times, our HDT pipeline is designed as a proactive and predominantly asynchronous process. The construction and validation of the PKG occur in the background and can tolerate moderate processing delays without degrading user experience. Latency-sensitive interactions are confined to direct conversational exchanges, whose response times are comparable to standard LLM-based assistants.

Future research may also explore extensions toward multi-user scenarios by enabling interactions or knowledge aggregation across multiple HDTs. For example, clusters of users with similar behaviors or contexts could be analyzed through the comparison or partial integration of their PKGs, enabling the extraction of collective insights while preserving personalized user models.

8. Conclusions and Limitations

This paper presented a pipeline for constructing Personal Knowledge Graphs within a Human Digital Twin architecture by integrating conversational extraction, multimodal contextualization, and ontology-driven validation. The approach operationalizes the Digital Brain as a coordinating service that transforms heterogeneous inputs, i.e. linguistic interactions, physiological measurements, calendar events, and sensor observations, into semantically grounded representations suitable for proactive reasoning. The proposed approach therefore represents a step toward more advanced Human Digital Twins. By enabling the transformation of heterogeneous user observations into structured knowledge, the proposed framework allows HDTs to move beyond monitoring and prediction toward supporting optimization and proactive responses.

Ablation results across 48 queries show that multimodal contextualization is essential rather than incremental: removing it reduces accuracy from 85% to 52%. The comparison between ontology-guided and ontology-free PKG construction further indicates that semantic validation is critical for reliability, reducing the false positive rate from 18% to 5%. These findings demonstrate that ontological grounding is a structural requirement for conversational knowledge extraction, enabling the system to filter spurious relationships that would otherwise compromise downstream inference. By combining symbolic validation with generative extraction, the proposed framework moves HDT architectures beyond passive monitoring toward proactive, knowledge-driven models capable of coherent reasoning over accumulated user information.

Several limitations point to future directions. The evaluation relies on synthetic scenarios rather than longitudinal interactions with real users, where linguistic variability and contextual ambiguity would provide a stronger test of generalization. The lack of external baselines reflects the absence of comparable end-to-end systems integrating multimodal contextualization with semantic validation, but future work should benchmark against emerging architectures as the field evolves. Finally, the use of a single embedding model for both triple alignment and query evaluation may introduce representational bias; independent embedding spaces or human-annotated validation protocols could strengthen the robustness of semantic assessments.

9. Acknowledgments

This work has been partially supported by the European Union - Next Generation EU under the Italian National Recovery and Resilience Plan (NRRP), Mission 4, Component 2, Investment 1.3, CUP C49J24000240004, partnership on

"Telecommunications of the Future" (PE00000001 - program "RESTART") and by the National Recovery and Resilience Plan (NRRP), Mission 4, Component 2, Investment 1.1, Call for tender No. 1409 published on 14.9.2022 by the Italian Ministry of University and Research (MUR), funded by the European Union – NextGenerationEU– Project Title "METATwin - Metaverse & Human Digital Twin: digital identity, Biometrics and Privacy in the future virtual worlds", – CUP J53D23015030001- Grant Assignment Decree No. 0001382 adopted on 01/09/2023 by the Italian Ministry of University and Research (MUR).

References

[1] J. Chen, et al., A revolution of personalized healthcare: Enabling human digital twin with mobile aigc, IEEE network (2024).

[2] I. Bisio, et al., Analysis of csi-based human activity recognition for contactless patients monitoring, in: GLOBECOM 2024-2024 IEEE Global Communications Conference, IEEE, 2024, pp. 438–443.

[3] R. Martinez-Velazquez, et al., Cardio twin: A digital twin of the human heart running on the edge, in: 2019 IEEE International Symposium on Medical Measurements and Applications (MeMeA), 2019, pp. 1–6. doi:10.1109/MeMeA.2019.8802162.

[4] H. Elayan, et al., Digital twin for intelligent context-aware iot healthcare systems, IEEE Internet of Things Journal 8 (2021) 16749–16757.

[5] Y. Lin, Y. Wang, M. Zhang, H. Cao, L. Ma, J. Wang, J. Gu, Towards efficient multi-domain knowledge fusion adaptation via low-rank reparameterization and noisy label learning: Application to source-free cross-domain fault diagnosis in iiot, IEEE Internet of Things Journal (2024) 1–1. doi:10.1109/JIOT.2024.3471410.

[6] K. Balog, T. Kenter, Personal knowledge graphs: A research agenda, in: Proceedings of the ACM SIGIR International Conference on the Theory of Information Retrieval (ICTIR), 2019.

[7] T. P. Raptis, C. Boldrini, M. Conti, A. Passarella, On the potential of an independent avatar to augment metaverse social networks, in: 2024 33rd International Conference on Computer Communications and Networks (ICCCN), 2024, pp. 1–9. doi:10.1109/ICCCN61486.2024.10637545.

[8] L. Serreli, M. Farina, M. Nitti, Next-gen iot: Human digital twins and large language models for proactive user engagement, IEEE Internet of Things Magazine (2025).

[9] W. Kritzinger, M. Karner, G. Traar, J. Henjes, W. Sihn, Digital twin in manufacturing: A categorical literature review and classification, IfacPapersOnline 51 (2018) 1016–1022.

[10] J. Lipkova, et al., Artificial intelligence for multimodal data integration in oncology, Cancer cell 40 (2022) 1095–1110.

[11] A. Vallée, Digital twin for healthcare systems, Frontiers in Digital Health 5 (2023) 1253050.

[12] P. Ruiu, M. Nitti, V. Pilloni, M. Cadoni, E. Grosso, M. Fadda, Metaverse & human digital twin: Digital identity, biometrics, and privacy in the future virtual worlds, Multimodal Technologies and Interaction 8 (2024) 48.

[13] J. Liu, et al., A hybrid deep-learning approach for complex biochemical named entity recognition, Knowledge-Based Systems 221 (2021) 106958.

[14] S. E. Yang Guo, Guohua Bai, Activity theory based ontology model for efficient knowledge sharing in ehealth, E-Health Telecommunication Systems and Networks 6 (2017) 31–45. doi:10.4236/etsn.2017.62003.

[15] C. Dias De Oliveira, et al., Exploring the feasibility of data-driven emotion modeling for human digital twins, in: Proceedings of the 16th International Conference on PErvasive Technologies Related to Assistive Environments, 2023, pp. 568–573.

[16] S. D. Okegbile, J. Cai, H. Zheng, J. Chen, C. Yi, Differentially private federated multi-task learning framework for enhancing human-to-virtual connectivity in human digital twin, IEEE Journal on Selected Areas in Communications 41 (2023) 3533–3547.

[17] J. Chen, C. Yi, S. D. Okegbile, J. Cai, X. Shen, Networking architecture and key supporting technologies for human digital twin in personalized healthcare: A comprehensive survey, IEEE Communications Surveys & Tutorials 26 (2023) 706–746.

[18] J. Chen, Y. Shi, C. Yi, H. Du, J. Kang, D. Niyato, Generative-ai-driven human digital twin in iot healthcare: A comprehensive survey, IEEE Internet of Things Journal 11 (2024) 34749–34773.

[19] P. Chakraborty, D. K. Sanyal, A comprehensive survey of personal knowledge graphs, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 13 (2023) e1513.

[20] W.-S. Kim, W. Kang, H.-J. Jeong, S. Lee, C. H. Song, S.-M. Choi, Conversational recommender systems based on extracting implicit preferences with large language models (2024).

[21] A. Tigunova, P. Mirza, A. Yates, G. Weikum, Exploring personal knowledge extraction from conversations with charm, in: Proceedings of the 14th ACM International Conference on Web Search and Data Mining, 2021, pp. 1077–1080.

[22] D. G. Vasques, S. O. Rezende, P. S. Martins, A semantic approach to uncovering implicit relationships in textual databases, in: 2018 XLIV Latin American Computer Conference (CLEI), IEEE, 2018, pp. 490–499.

[23] J. Dagdelen, A. Dunn, S. Lee, N. Walker, A. S. Rosen, G. Ceder, K. A. Persson, A. Jain, Structured information extraction from scientific text with large language models, Nature communications 15 (2024) 1418.

[24] J. Quan, S. Zhang, Q. Cao, Z. Li, D. Xiong, Risawoz: A large-scale multi-domain wizard-of-oz dataset with rich semantic annotations for task-oriented dialogue modeling, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 930–940.

[25] D. Xu, W. Chen, W. Peng, C. Zhang, T. Xu, X. Zhao, X. Wu, Y. Zheng, Y. Wang, E. Chen, Large language models for generative information extraction: A survey, Frontiers of Computer Science 18 (2024) 186357.

[26] L. Serreli, C. Marche, M. Nitti, Reducing data volume in news topic classification: Deep learning framework and dataset, IEEE Open Journal of the Computer Society 6 (2025) 153–164. doi:10.1109/OJCS.2024.3519747.

[27] B. Zhang, H. Soh, Extract, define, canonicalize: An llm-based framework for knowledge graph construction, in: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, 2024, pp. 9820–9836.

[28] Y. Lairgi, L. Moncla, R. Cazabet, K. Benabdeslem, P. Cléau, itext2kg: Incremental knowledge graphs construction using large language models, in: International Conference on Web Information Systems Engineering, Springer, 2024, pp. 214–229.

[29] S. Carta, A. Giuliani, L. Piano, A. S. Podda, L. Pompianu, S. G. Tiddia, Iterative zero-shot llm prompting for knowledge graph construction, arXiv preprint arXiv:2307.01128 (2023).

[30] P. Vossen, S. B. Santamaría, L. Bajčetić, T. Belluci, Extracting triples from dialogues for conversational social agents, arXiv preprint arXiv:2412.18364 (2024).

[31] A. S. Kwak, M. Alexeeva, G. Hahn-Powell, K. Alcock, K. McLaughlin, D. McCorkle, G. McNunn, M. Surdeanu, Information extraction from conversation transcripts: Neuro-symbolic vs. llm, arXiv preprint arXiv:2510.12023 (2025).

[32] C. Jiang, B. Qi, X. Hong, D. Fu, Y. Cheng, F. Meng, M. Yu, B. Zhou, J. Zhou, On large language models' hallucination with regard to known facts, in: Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), 2024, pp. 1041–1053.

[33] H. T. Mai, C. X. Chu, H. Paulheim, Do llms really adapt to domains? an ontology learning perspective, in: International Semantic Web Conference, Springer, 2024, pp. 126–143.

[34] Q. Chen, H. Sun, H. Liu, Y. Jiang, T. Ran, X. Jin, X. Xiao, Z. Lin, H. Chen, Z. Niu, An extensive benchmark study on biomedical text generation and mining with chatgpt, Bioinformatics 39 (2023) btad557.

[35] I. Al-Zyoud, F. Laamarti, X. Ma, D. Tobon, A. El Saddik, Towards a machine learning-based digital twin for non-invasive human bio-signal fusion, Sensors 22 (2022) 9747.

[36] X. Hou, Y. Zhao, S. Wang, H. Wang, Model context protocol (mcp): Landscape, security threats, and future research directions, arXiv preprint arXiv:2503.23278 (2025).

[37] J. Yan, D. Li, Q. Yang, Llm-trafficbrain: An information-centric framework for dynamic signal control with large language models, in: 2025 IEEE 26th China Conference on System Simulation Technology and its

Applications (CCSSTA), IEEE, 2025, pp. 252–256.
[38] M. Mohammadi, A. A. Atashin, W. Hofman, Y. Tan, Comparison of ontology alignment systems across single matching task via the mcnemar's test, ACM Transactions on Knowledge Discovery from Data (TKDD) 12 (2018) 1–18.
[39] J. Chen, O. Mashkova, F. Zhapa-Camacho, R. Hoehndorf, Y. He, I. Horrocks, Ontology embedding: a survey of methods, applications and resources, IEEE Transactions on Knowledge and Data Engineering (2025).
[40] D. Gromann, T. Declerck, Comparing pretrained multilingual word embeddings on an ontology alignment task, in: Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018), 2018.
[41] M. Taboada, D. Martinez, M. Arideh, R. Mosquera, Ontology matching with large language models and prioritized depth-first search, Information Fusion (2025) 103254.
[42] M. A. Rodriguez, J. H. Watkins, J. Bollen, C. Gershenson, Using rdf to model the structure and process of systems, in: Unifying Themes in Complex Systems VII: Proceedings of the Seventh International Conference on Complex Systems, Springer, 2012, pp. 222–230.
[43] L. Perez-Beltrachini, C. Gardent, Learning embeddings to lexicalise rdf properties, in: Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics, 2016, pp. 219–228.
[44] J. Collio, A. Aguilera, I. Dongo, Towards an enrichment of ontologies inferred from rdf documents, IEEE Access (2024).
[45] B. Zhao, K. Peng, K. Zhang, H. Sun, Z. Tu, D. Chu, Serflow: A multi-stage service-enhanced mechanism for workflow applications in cpss with end-edge-cloud collaboration, IEEE Internet of Things Journal (2025).
[46] Z. Zhao, M. Zhang, J. Chen, T. Qu, G. Q. Huang, Digital twin-enabled dynamic spatial-temporal knowledge graph for production logistics resource allocation, Computers & Industrial Engineering 171 (2022) 108454.
[47] K. Peng, B. Zhao, C. Ling, M. Bilal, X. Xu, J. J. Rodrigues, Tofds: A two-stage task execution method for fake news in digital twin-empowered socio-cyber world, IEEE Transactions on Computational Social Systems 11 (2023) 5178–5188.
[48] C. Ramonell, R. Chacón, H. Posada, Knowledge graph-based data integration system for digital twins of built assets, Automation in Construction 156 (2023) 105109.
[49] O. Chukhno, N. Chukhno, G. Araniti, C. Campolo, A. Iera, A. Molinaro, Learning-powered migration of social digital twins at the network edge, Computer Communications 226 (2024) 107918.
[50] C. Wang, J. Qiao, X. Huang, S. Song, H. Hou, T. Jiang, L. Rui, J. Wang, J. Sun, Apache iotdb: A time series database for large scale iot applications, ACM Transactions on Database Systems 50 (2025) 1–45.
[51] M. Salman, A. Haller, S. J. R. Méndez, U. Naseem, Tiny but mighty: A crowdsourced benchmark dataset for triple extraction from unstructured text, in: Proceedings of the 20th Joint ACL-ISO Workshop on Interoperable Semantic Annotation@ LREC-COLING 2024, 2024, pp. 71–81.
[52] K. Kolpashnikova, S. Flood, O. Sullivan, L. Sayer, E. Hertog, M. Zhou, M.-Y. Kan, J. Suh, J. Gershuny, Exploring daily time-use patterns: Atusx data extractor and online diary visualization tool, Plos one 16 (2021) e0252843.
[53] A. Zaveri, A. Rula, A. Maurino, R. Pietrobon, J. Lehmann, S. Auer, Quality assessment for linked data: A survey, Semantic Web 7 (2016) 63–93.
[54] S. Choi, Y. Jung, Knowledge graph construction: Extraction, learning, and evaluation, Applied Sciences 15 (2025) 3727.
[55] C. Yi, J. Cai, Z. Su, A multi-user mobile computation offloading and transmission scheduling mechanism for delay-sensitive applications, IEEE Transactions on Mobile Computing 19 (2019) 29–43.
[56] Y. Yang, Y. Shi, C. Yi, J. Cai, J. Kang, D. Niyato, X. Shen, Dynamic human digital twin deployment at the edge for task execution: A two-timescale accuracy-aware online optimization, IEEE Transactions on Mobile Computing 23 (2024) 12262–12279.