Talking Buildings: Interactive Human-Building Smart-bot for Smart Buildings

Devmane Suhas[1✉][0009−0007−4226−5567], Omer Rana[1][0000−0003−3597−2646], Simon Lannon[2][0000−0003−4677−7184], and Charith Perera[1][0000−0002−0190−3346]

[1] School of Computer Science and Informatics, Cardiff University, Cardiff, UK
[2] Welsh School of Architecture, Cardiff University, Cardiff, UK
{DevmaneSP1,RanaOF,Lannon,PereraC}@cardiff.ac.uk

Abstract. Conventional conversational agents have a limited ability to respond to different user intents when interacting with smart buildings. The uniqueness of each building, coupled with the heterogeneity of built environments, makes it challenging to adapt communication methods universally. A possible solution is to develop a conversational agent capable of understanding physical, logical, and virtual assets in the built environment, with the aim of establishing a standardised method of human-building communication. Current smart building ontologies and metadata description schemas aim to give smart buildings a common language for the rapidly growing number of devices in smart buildings. This research paper focuses on developing a comprehensive smart building framework that integrates chatbot-driven natural language interactions into smart buildings using the SPARQL query language to query the smart building Knowledge Base (KB) and interact with buildings using a chatbot. An environmental sensor network testbed was set up using an ontology representing a smart building to evaluate the answer to the question. We have used transformer-based machine learning (ML) models to translate the natural language (NL) queries into SPARQL queries and summarise combined SPARQL and natural language queries, which produced promising performance. By integrating chatbots into smart building systems on the edge, users can interact in natural language, provide real-time information, and detect potential threats without the need for specialised knowledge. Our future work will be to extend this model to support heterogeneous building types represented by smart building ontologies. The source code and data sets are publicly available. [3]

Keywords: Semantic Web · Internet-of-Things (IoT) · Smart Buildings · Human-Building Interaction (HBI)

1 Introduction

Human Building Interaction (HBI), an interdisciplinary field, studies the complex interplay between computer technology and the built environment/smart buildings [2]. This field of study is concerned with dynamic interactions between

[3] https://github.com/suhasdevmane/abacws-chatbot

humans and computers and between people and their physical environments. Previous studies suggest that in addition to identifying the barriers to their widespread adoption, it is necessary to investigate the communication interfaces between buildings or their services [1]. Smart buildings have recently attracted significant attention, mainly due to the desire to improve operational effectiveness, occupant comfort, and energy efficiency [23]. Technological developments in natural language processing (NLP) and knowledge representation have sparked the creation of novel approaches that make use of chatbots. Furthermore, the rise of the Internet of Things (IoT) has changed how devices are connected in the built environment and has opened up access to a massive amount of data.

SmartThings (Samsung), Google Home (Google), HomeKit (Apple), Alexa (Amazon), and other platforms use NLP and AI technologies for user intent recognition, entity extraction, and response generation [7]. These systems rely on patterns derived from large datasets, processed dynamically or pre-trained in the cloud, without knowledge of smart building environments. Since smart buildings vary in system and sensor, data is stored in different databases, making it difficult for traditional cloud-based chatbots to access and interact with these environments effectively. In addition, these agents lack domain-specific knowledge, leading to inconsistent query results in diverse smart buildings. A uniform approach to understanding user intent across different smart environments is unfeasible; each building requires a unique knowledge base to ensure accurate responses. Developing chatbots that understand both user intent and the built environment, leveraging knowledge representation techniques such as ontologies, could facilitate efficient communication with heterogeneous smart buildings. Aligned with contemporary advancements in the field, our research focuses on addressing the following objectives, ensuring accessibility for any smart building user, regardless of expert knowledge, to engage in meaningful 'talk' with the building:

1. What elements and architectural considerations are essential to enable natural language interactions with smart buildings for diverse objectives?
2. What components are required for the development of adaptable algorithms and applications within the smart buildings knowledge base, incorporating real-time automated reasoning on data that can be deployed across a variety of building types while inherently supporting application-specific functions?

Smart buildings need a unified metadata description framework applicable to all building types to facilitate human-building interactions, providing a comprehensive knowledge representation approach. We utilised Brick [5], a widely adopted semantic building metadata method, to develop and evaluate responses to user questions by translating natural language (English in our case) into SPARQL Protocol and RDF Query Language (SPARQL) using a Transformers-based ML model [39]. Integrating the BrickSchema with reasoning applied to the building ontology enables a semantic understanding of all physical, logical, and virtual assets within the building, clarifying their interrelationships. Our framework includes the Rasa conversational agent (chatbot), based on Natural

Language Understanding (NLU), and SPARQL, a query language for semantic web databases, to efficiently access and process data from the building's knowledge graph. The NLU component handles intent classification, entity extraction, and response retrieval, enabling intelligent interactions between users and smart buildings.

In brief, the contributions of our paper are summarised as follows

- We propose a framework to promote the field of HBI and smart building technologies by providing an intuitive and effective method of human-building conversation through chatbots, ML models with domain knowledge, and smart home description using ontologies.
- We introduced two datasets: one for training the ML models and another containing environmental data from a smart building testbed for natural language conversation.
- We introduced the trained ML models based on transformers having domain knowledge to fine-tune the new building ontologies for human-building conversation for multiple intents.

The organisational structure of this paper is as follows. Section 2 provides a comprehensive overview of the current background and related work in HBI, focusing on the use of conversational agents employing ontology and ML models used for question-answering and text summarisation. Section 3, the overall system architecture of the proposed framework is detailed, providing insight into its structural components and functionalities. Finally, Section 4 describes the dataset, ontologies used, and the implementation and experimental setup of this framework. Finally, Section 5 critically evaluates the current framework from different perspectives, assessing its effectiveness, limitations and potential areas for improvement.

2 Related Work

There has been a surge in studies on user behaviour, interactive design, and smart devices to support diverse and interdisciplinary research on smart buildings [44] among researchers. To enhance human-building communication, recent studies have adopted interactive design principles. A voice assistant was examined to see if the differences in modality (i.e., voice vs. text) and device (i.e., smartphone vs. smart home device) affect user perceptions when users attempt to retrieve sensitive health information from voice assistants. It has been seen that, among all the types above, conversational agents are preferred by users [12]. Others explored design needs, user expectations[14,16], communication channels and authentication[19], multiuser experience and design recommendations[24] spatial consideration using Building Information Modeling (BIM) [29], household social needs [35], future directions and challenges to be addressed by chatbot research[15] to better interact in built environments. Smart buildings, equipped with diverse components and sensors, provide valuable information to stakeholders and facilitate user interactions through systems like Building Management Systems (BMS) [36] and Building Automation Systems (BAS) [8] which requires expert knowledge to execute tasks within the built environment.

Ontologies have become essential for describing domain-specific knowledge, particularly in smart buildings, where various efforts have focused on defining entities and relationships to solve interoperability issues [32,22,9,21,13,10,11]. Ontologies such as Semantic Sensor Networks (SSN) and Sensor, Observation, Sample, and Actuator (SOSA) [43,18] offer frameworks for modelling sensor data. Brick [5], an open-source Resource Description Framework (RDF) based schema, was introduced to describe the structure and functionality of buildings, emphasising semantic descriptions of physical, logical, and virtual assets and their interrelationships. Brick prioritises completeness, expressivity, usability, consistency, and extensibility. We utilised Brick to create an ontology tailored to describe a smart building sensor network. Unlike Web Ontology Language (OWL), which is part of the W3C Semantic Web stack and uses Description Logic (DL), Brick is built on RDF and RDF-Schema (RDFS) semantics without employing DL.

Chatbot architectures suggest that formalising rational components (reasoning and NLP capabilities) and intuitive components (semantics) is crucial to improving chatbot knowledge bases to support human-like conversation [3]. Recently, ontology-based chatbots have gained popularity in domains, demonstrating effective communication with users. Examples include a tutoring chatbot for students [30], an e-commerce assistant [40], and MediBot, a medical assistant [4], all using ontologies alongside other components. Ontology-driven chatbots utilise linguistic rule-based systems and syntactic ambiguity resolutions to accurately detect user intent [34]. ML models are used to automate various tasks. Transformer-based models are used for patient monitoring in smart homes [25], human activity recognition (HAR) [27,20], security [41], and energy efficiency [17]. Recently, a Bidirectional Encoder Representations from Transformers (BERT) model has been used to answer questions in knowledge graphs [38]. A Text-to-Text Transfer Transformer (T5) based model is trained for SPARQL to generate NL questions for knowledge-based conversational applications, which can be used to generate SPARQL using datasets such as LC-QuAD 2.0 and ParaQA, CSQA, WebNLG-QA [26]. The Bidirectional Auto-Regressive Transformers (BART) based model is trained to generate SPARQL questions to interact with the DBLP database using the DBLP-QuAD dataset [42]. BART excels in sequence-to-sequence tasks such as summarisation, translation, and question-answering by preserving key details, while T5's unified text-to-text approach enhances performance across these tasks [26]. Integrating these models with a knowledge base can significantly improve human interaction with smart buildings.

3 System Architecture

Figure 1 illustrates the system architecture with its main components: Graphical User Interfaces (GUI) and back-end processes. The GUI, known as services, enables user communication with smart buildings and consists of interconnected services necessary to complete the dialogue. The back-end processes handle user intents, process inputs, and generate output through custom actions and ML models. Table 1 explains the components attached and their role in fulfilling the purpose of the human-building conversation.

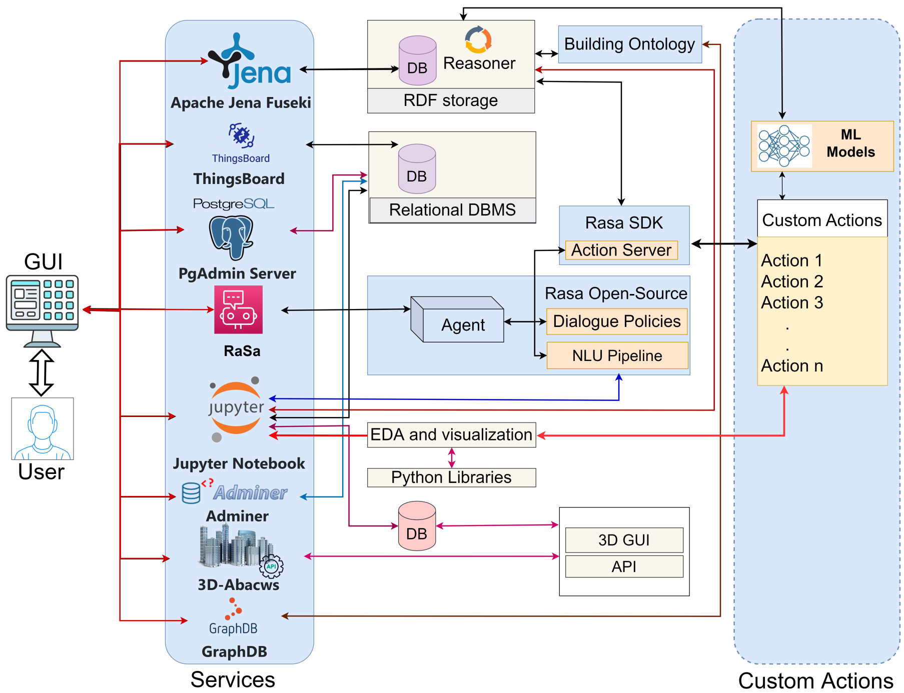

Fig. 1. Overview of system architecture

Table 1. Element requirements and selections for human-building conversations.

Element Required	Selected Element	Application
Semantic description of Smart building	BrickSchema ontology	To semantically describe physical, logical and virtual assets and their relationships.
Reasoning on ontology	shacl, owlrl, vbis	To facilitate the validation of RDF data against predefined shapes, ensuring compliance with constraints.
A language to talk to ontology	SPARQL	To retrieve information from RDF databases in the context of ontologies.
RDF triple store	Apache Jena Fuseki	To handle SPARQL queries and keep requests and responses in a manner consistent with the ontology's semantic structure.
Conversational Chatbot	RaSa Open Source	To interact with the smart building system using natural language that is connected to various services.
Timeseries Database	PostgreSQL	To store large volumes of data from our sensor network with unique IDs.
IoT platform	ThingsBoard	To gather, handle, display, and evaluate data from IoT devices that are connected to PostgreSQL and the PgAdmin server for data management and administration.
Natural language to SPARQL translation	BART (bart-base), T5 (t5-base)	To convert the natural language question received from Rasa to the SPARQL query.
SPARQL summarisation	T5 (t5-base)	To summarise the response received from the knowledge base and database to a meaningful summary for the user.
Training Datasets	Custom	To train the transformer model for NL to SPARQL translation and summarisation of received data from database and ontology.
Smart Building Visualization	Abacws3D	To visualise the live sensor data and monitor buildings health using React and TypeScript.
Programming language	Python, Flask, Node, Docker, Jupyter Notebook	To avoid dependency conflicts at end-user, Data Analysis (EDA) and Visualization, Perform custom application using portable application.

3.1 Services

We adopted the BrickSchema as our building metadata schema due to its structural advantages, effectively addressing the limitations of previous schemas. This choice ensures a well-defined ontology that is maintainable within set specifications. For reasoning within the BrickSchema ontology, we utilised its supported reasoning profile, streamlining the integration of reasoning mechanisms. SPARQL was used for querying and interacting with the data, connecting with the Apache Jena-Fuseki SPARQL endpoint that holds the building ontology, receives SPARQL queries from the Rasa action server, and incorporates reasoners to maintain standards. We used Rasa Open Source, a popular framework for building AI assistants in Python that supports multiple platforms. It manages training data, including NLU data, stories, and rules, in YAML format, with definitions for forms and responses within the domain. Rasa uses regular expressions, lookup tables, and synonyms to train NLUs on categorised user utterances, employs step-based stories composed of user messages and actions, and uses an ML pipeline for intent classification and custom actions.

The action server executes custom actions based on user queries, running any Python code, such as database queries and API calls. The bot's actions and responses are followed based on the stories. The dialogue model predicts the next step based on user messages, while rules train the assistant's dialogue management model. The processing pipeline, defined in *config.yml*, runs a sequence of components to efficiently handle incoming messages. To enhance data acquisition and evaluation, we deployed an environmental sensor network integrated with an IoT platform in an academic building. Sensor data is transmitted to a PostgreSQL database, contributing to a dynamic dataset with autogenerated IDs and keys, which are added to the building ontology for external data reference. For data exploration, visualisation, testing, debugging, and administration, we integrated tools such as Adminer, PgAdmin, and GraphDB with the PostgreSQL database and the RDF store. A 3D visualisation service provides a visual representation of the environmental sensors, retrieving data via an API. In addition, an IoT platform virtualises devices and manages data storage, administration, actions, and visualization of data within the local network.

3.2 Machine Learning Models and Custom Actions

Figure 2 shows the flow of a natural language conversation between a user and a smart building, where a chatbot mediates between the user and the building by processing intentions and providing answers. The conversation agent first receives the user's question and processes it through the NLU pipeline to determine the intent. If the intent is out of scope, it is classified as a custom action. Custom actions are sets of operations that fulfil user intents. The custom action script considers the user's input and sends it to the BART or T5 model for tokenisation. The input text is converted into word embeddings and translated into a SPARQL query. The query is sent to the RDF knowledge base to fetch the required data. If the answer includes data stored in a database, it retrieves the

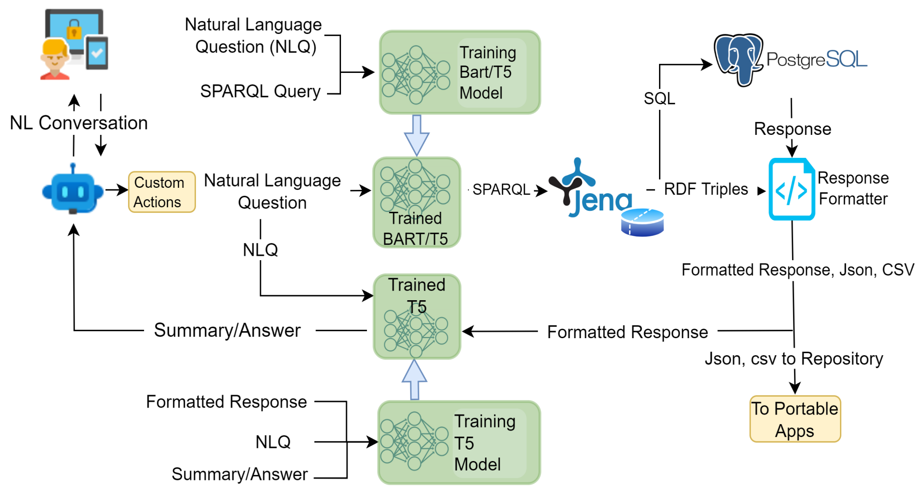

Fig. 2. Machine learning model and custom actions flow

associated data using SQL queries with unique UUIDs. The responses from both the RDF storage and the PostgreSQL database are converted into well-formatted RDF triples. These triples are stored for data analysis using portable applications in CSV and JSON format. T5 model receives this formatted response, which summarises the input in a readable format based on the user's Natural Language Question (NLQ) and formatted SPARQL response. The summarised response is then sent to the chatbot interface.

Response Formatter : The system includes a sophisticated response formatter script designed to process responses from both the SPARQL endpoint and the PostgreSQL database. This script efficiently handles RDF data by removing URLs and appending the respective PREFIX, ensuring a clean and readable output. Additionally, it processes database responses to produce a consistent and understandable format suited for transformer models trained on our dataset. This formatting ensures that the data are presented in a way that maximises the effectiveness of the ML models, thereby enhancing the overall performance and accuracy of the system in understanding and generating responses.

Natural Language Translation and summarisation : Text-to-text frameworks in transfer learning have shown remarkable capabilities for natural language generation, translation, and comprehension. ML T5 models are trained using natural language questions, formatted SPARQL query responses, and summarisation text. The training dataset contains more than four thousand examples to construct the summary from the user input question, and the formatted SPARQL response. BART is an ML model trained using natural language questions and SPARQL query pairs to construct a SPARQL query.

4 Implementation

The framework for smart buildings integrates a Rasa chatbot, a SPARQL endpoint with a local database, and a building ontology hosted in Apache Jena Fuseki. This setup allows for seamless natural language interaction, translating user intents into SPARQL queries executed on the local database. The ontology represents smart building elements. The SPARQL query retrieves sensor information based on a Universally Unique Identifier (UUID) within a semantic description in smart building ontology. It selects properties like sensor, external reference, UUID, database, label, and connection string, using RDF triples to navigate relationships via the Brick ontology. The query filters sensors by UUID and links them to external references associated with time-series data stored in a database, extracting the data points, database type, label, and connection string. This SPARQL query effectively extracts structured data and ML models summarise the responses and sends to the rasa actions for the user.

Dataset Structure To successfully train a model, a large-quantity bilingual parallel corpus is needed. We have used a custom-created training dataset with a number of 5,470 sets for training the T5 and BART models, including natural language questions, the corresponding SPARQL query, response, and explanation, as shown in the example below:

```
{
  "ID": 4050,
  "en": "How does this smart building monitor and maintain air quality
      in the east zone?",
  "sparql": "SELECT ?sensor WHERE { ?sensor a brick:Air_Quality_Sensor
      ; brick:hasLocation bldg:east-Zone . }",
  "response": "sensor: bldg:airq5.04, bldg:airq5.05, bldg:airq5.34,
      bldg:airq5.25, bldg:airq5.28",
  "explanation": "Sensors located in the east zone monitor air
      quality."
}
```

Ontology Structure The ontology defines entities such as buildings, sensors, and their relationships using the brickschema. It contains 4899 RDF triples before any reasoning is performed. It has information about all the floors, sensor devices and their UUIDs, where their information is stored in the database. An example snippet of an ontology is as follows.

```
@prefix bldg: <http://abacwsbuilding.cardiff.ac.uk/abacws#> .
@prefix brick: <https://brickschema.org/schema/Brick#> .
bldg:Abacws a rec:Building, rec:School ;
    rec:address "Cardiff, UK" ;
    rec:architectedBy "Cardiff University" .
bldg:airq5.01 a brick:Air_Quality_Sensor ;
    rdfs:label "Room Air Quality Sensor" ;
    ref:hasExternalReference [
        ref:hasTimeseriesId "478abf30-7db2-11ee-b0f3-69bd975277c1"
    ] .
```

Table 2. Natural language conversation flow between human-building

<table>
<tr><td>Question</td><td>How this smart building monitor and maintain the air quality in the east zone?</td></tr>
<tr><td>SPARQL Query Generated</td><td>SELECT ?sensor
WHERE {
?sensor a brick:Air_Quality_Sensor ;
brick:hasLocation bldg:east-Zone . }</td></tr>
<tr><td>Formatted SPARQL Response</td><td>sensor: bldg:airq5.04, sensor: bldg:airq5.05, sensor: bldg:airq5.34, sensor: bldg:airq5.25, sensor: bldg:airq5.28</td></tr>
<tr><td>Received Answer</td><td>The output from the SPARQL endpoint lists several air quality sensors. Sensor airq5.04, Sensor airq5.05, Sensor airq5.34, Sensor airq5.25, Sensor airq5.28 are sensors located in the east zone of the building to monitor air quality.</td></tr>
<tr><td>Question</td><td>what is the average temperature level in east-zone on the Floor 5 in the building between 1st May 2023 and 30 January 2024?</td></tr>
<tr><td>SPARQL Query Generated</td><td>SELECT ?sensor ?timeseries_id ?timeseries_key_id
WHERE {
?sensor a brick:Temperature_Sensor ;
brick:isPointOf bldg:east-Zone ;
ref:hasExternalReference ?ref .
?ref ref:hasTimeseriesId ?timeseries_id ;
ref:hasTimeseriesKeyId ?timeseries_key_id .
}</td></tr>
<tr><td>Formatted SPARQL Response</td><td>sensor: bldg:temp5.05, timeseries_id: 78df12c0-7db2-11ee-b0f3-69bd975277c1, timeseries_key_id: 133, sensor: bldg:temp5.34, timeseries_id: 9586a700-83f5-11ee-a992-8978af4232d8, timeseries_key_id: 133, sensor: bldg:temp5.25, timeseries_id: 6694ad70-83f5-11ee-a992-8978af4232d8, timeseries_key_id: 133, sensor: bldg:temp5.28, timeseries_id: 759e7850-83f5-11ee-a992-8978af4232d8, timeseries_key_id: 133</td></tr>
<tr><td>Received Answer</td><td>Overall Average Temperature: {overall_average}</td></tr>
<tr><td>Question</td><td>Compare CO2 levels recorded by sensors in the north-west and south-west zones between 10/11/2023 to 29/02/2024.</td></tr>
<tr><td>SPARQL Query Generated</td><td>SELECT ?northWestTimeseriesId ?southWestTimeseriesId
WHERE {
OPTIONAL {
?sensor a brick:CO2_Sensor ;
brick:hasLocation bldg:north-west-Zone ;
ref:hasExternalReference [
ref:hasTimeseriesId ?northWestTimeseriesId] . }
OPTIONAL {
?sensor a brick:CO2_Sensor ;
brick:hasLocation bldg:south-west-Zone ;
ref:hasExternalReference [
ref:hasTimeseriesId ?southWestTimeseriesId] . } }</td></tr>
<tr><td>Formatted SPARQL Response</td><td>northWestTimeseriesId:478abf307db211eeb0f369bd975277c1,
southWestTimeseriesId:478abf307db211eeb0f369bd975277c1,
southWestTimeseriesId:9f44f0107db211eeb0f369bd975277c1....</td></tr>
<tr><td>Received Answer</td><td>Overall Average CO2 in north-west is : {overall_average} and south-west is {overall_average}</td></tr>
</table>

Table 2 illustrates how a user-entered natural language query is converted to a SPARQL query and how the formatter formats the response returned by the SPARQL database. The T5 model then generates the response for Rasa actions.

4.1 Experimental Setup

All experiments were conducted on an Amazon EC2 instance configured with a Tesla T4 GPU, enabling faster convergence. The g4dn.4xlarge instance provided sufficient computational power to handle the intensive training and evaluation processes required for the models used. The machine setup is outlined in Table 3.

Table 3. Specifications of the training machine.

Component	Specification
Product	g4dn.4xlarge
CPU	Intel Xeon Platinum 8259CL, 8 cores (16 threads) @ 2.50GHz
Memory	64 GiB DDR4
Storage	150 GiB NVMe SSD
GPU	1 x NVIDIA Tesla T4

For our experiments, we leveraged the BART and T5 models, which are highly regarded for their sequence-to-sequence capabilities. BART, known for combining bidirectional and autoregressive transformers, is particularly effective for tasks like text generation. On the other hand, T5 treats all NLP tasks as text-to-text, making it versatile across various natural language processing domains. Both models were used to convert natural language questions into SPARQL queries, while T5 is used to summarise the output of SPARQL queries and database answers, as T5 showed promising performance for text summarisation of various datasets [37].

To ensure the models could understand domain-specific language, we enhanced the tokeniser by adding tokens from the Abacws smart home ontology and the BrickSchema ontology (v1.4). These tokens, along with RDF triples and SPARQL query tokens, were processed using the ‘rdflib‘ library and incorporated into the training pipeline. This preprocessing step allowed the models to effectively map natural language queries to their corresponding SPARQL queries within the smart building domain. Table 4 presents the evaluation metrics for BART and T5 across different tasks. We observed the performance for tasks such as SPARQL query generation and output summarisation.

Table 4. Model Evaluation Metrics

Model Name	Task	Eval Loss	Runtime (s)	Samples/s	Steps/s
BART-NL2SPARQL	SPARQL Generation	0.0755	2.2169	123.145	31.125
T5-NL2SPARQL	SPARQL Generation	0.0019	3.3988	15.888	2.06
T5-SPARQL2QA	summarisation	0.0838	22.0205	12.398	1.589

As shown in Table 4, the models performed well on the evaluation dataset, which comprised 10% of the total data. The training time ranged from 3 to 4 hours, depending on the model and dataset size. For both models, we trained using the Seq2Seq framework, optimising key parameters such as a learning rate of 2e-5, a batch size of 1, and 20 training epochs. Evaluation metrics such as loss, runtime, samples per second, and steps per second were recorded to track the model performance.

The dataset used for training consisted of 5470 samples, with an 80-20 train-test split. 10% of the data was reserved for validation to assess model performance throughout the training process. The RDF triples (4899 triples) and tokens were derived from the BrickSchema ontology and Abacws smart home ontology. These ontologies were verified using Protégé, an open-source ontology editor, to ensure that the SPARQL queries generated were both syntactically and semantically correct.

5 Result and Discussion

To evaluate the performance of our models, we used three common metrics for natural language generation tasks: BLEU[31], ROUGE[28], and METEOR[6]. These metrics were chosen because of their widespread use in the evaluation of text generation models and their ability to provide a comprehensive view of model performance [33].

- BLEU: The Bilingual Evaluation Understudy (BLEU) score measures the precision of n-grams between the generated and reference texts. Higher BLEU scores indicate better performance.
- ROUGE: Recall-Oriented Understudy for Gisting Evaluation (ROUGE) measures the overlap of n-grams and the longest common subsequence between the generated and reference texts. We report ROUGE-L scores.
- METEOR: The Metric for Evaluation of Translation with Explicit ORdering (METEOR) evaluates generated text by considering synonyms, stemming, and paraphrasing. Higher METEOR scores indicate better performance.

We evaluated the BART and T5 models trained for SPARQL query generation and the T5 model trained for text summarisation. The results are summarised in Table 5 for the evaluation dataset used.

Table 5. Models performance on SPARQL generation and text summarisation

Task	**Model**	**BLEU**	**ROUGE-L**	**METEOR**	**Res. Time (ms)**
SPARQL Generation	BART	0.6078	0.7054	0.6089	1500
	T5	**0.6432**	**0.7102**	**0.6423**	1350
Text summarisation	T5	0.6232	0.6574	0.6012	1450

The evaluation results show that both the BART and T5 models perform well on the SPARQL query generation task, with T5 slightly outperforming BART in terms of BLEU, ROUGE-L and METEOR scores. For the text-summarisation T5 also achieved strong scores, indicating its effectiveness in summarising the outputs. The system has demonstrated a commendable degree of success in fulfilling user intentions for the constructed building ontology. This success can be attributed to several factors that collectively enhance its performance.

Smart Building Ontology: A general and expandable ontology for smart buildings provides the framework for representing various components like sensors, actuators, rooms, and environmental parameters.

NLU: NLU capabilities in the Rasa chatbot enable intuitive and user-friendly interactions by effectively mapping user intents to ontology searches, converting user input into executable commands.
SPARQL Query Engine: The resilient SPARQL query engine swiftly executes dynamic queries against the local SPARQL database, providing accurate and timely results. Its real-time data processing capabilities enhance system effectiveness in managing streaming sensor data and performing automatic reasoning based on ontology.
Machine Learning Models: Training advanced ML models, notably T5 and BART, on large datasets improves human-building interaction by enabling the system to comprehend and construct SPARQL queries from NL inputs and summarise query results efficiently.

Using BrickSchema, we ensure the ontology is structured to support smart building components like sensors, actuators, and environmental controls via custom actions. The ontology is both flexible and extensible, enabling consistent performance, accurate queries, and smooth integration with other architectures, while allowing future updates without disrupting current functions. To ensure reliability, we continuously refine and validate the ontology. The current machine learning models trained on BrickSchema can be fine-tuned for different schemas and datasets, allowing adaptability. We're also improving accuracy by training models on larger datasets by expanding current datasets. Key priorities include monitoring response times under heavy loads and ensuring data persistence. Our iterative improvements, guided by analytics and user feedback, will keep the framework robust. Regular documentation updates will help users adapt and expand the system as needs evolve.

6 Conclusion

The system has successfully demonstrated the ability to fulfil user intentions across various building ontologies using a generic and extensible ontology, a robust SPARQL query engine, and advanced NL understanding capabilities integrated within the Rasa chatbot. In achieving our objectives, we have successfully enabled NL interactions with smart buildings by developing an adaptable framework that supports both NL query translation and semantic understanding through SPARQL queries. The framework integrates smart building data retrieval and reasoning capabilities by incorporating ML models to enable real-time, automated reasoning on building data, ensuring its deployment across various building types. Additionally, the framework offers flexibility for diverse user needs, allowing the system to handle multiple intents and adapt to a wide range of building environments, as demonstrated by our testbed setup and the successful translation of NL inputs into building-specific queries. The incorporation of T5 and BART models trained on the large datasets further enhances the system's capacity for efficient human-building interactions. These efforts ensure that the system remains a leading solution in smart building management, offering a versatile and user-friendly interface for diverse applications.

Acknowledgments. This work is supported in part by the EPSRC PETRAS (Grant No. EP/S035362/1), the GCHQ National Resilience Fellowship, and ARCCA at Cardiff University.

References

1. Alavi, H.S., Churchill, E., Kirk, D., Nembrini, J., Lalanne, D.: Deconstructing human-building interaction. Interactions **23**(6), 60–62 (2016)
2. Alavi, H.S., Churchill, E.F., Wiberg, M., Lalanne, D., Dalsgaard, P., Schieck, A.F.G., Rogers, Y.: Introduction to human-building interaction (HBI): Interfacing HCI with architecture and urban design. ACM Trans. Comput. Interact. **26**(2) (2019)
3. Augello, A., Pilato, G., Vassallo, G., Gaglio, S.: Chatbots as Interface to Ontologies, pp. 285–299. Springer International Publishing, Cham (2014)
4. Avila, C.V.S., Calixto, A.B., Rolim, T.V., Franco, W., Venceslau, A.D., Vidal, V.M., Pequeno, V.M., De Moura, F.F.: Medibot: An ontology based chatbot for Portuguese speakers drug's users. ICEIS 2019 - Proc. 21st Int. Conf. Enterp. Inf. Syst. **1**, 25–36 (2019)
5. Balaji, B., Bhattacharya, A., Fierro, G., Gao, J., Gluck, J., Hong, D., Johansen, A., Koh, J., Ploennigs, J., Agarwal, Y., Bergés, M., Culler, D., Gupta, R.K., Kjærgaard, M.B., Srivastava, M., Whitehouse, K.: Brick: Metadata schema for portable smart building applications. Appl. Energy **226**(February), 1273–1292 (2018)
6. Banerjee, S., Lavie, A.: METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In: Goldstein, J., Lavie, A., Lin, C.Y., Voss, C. (eds.) Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization. pp. 65–72. Association for Computational Linguistics, Ann Arbor, Michigan (Jun 2005)
7. Berdasco, A., López, G., Diaz, I., Quesada, L., Guerrero, L.A.: User Experience Comparison of Intelligent Personal Assistants: Alexa, Google Assistant, Siri and Cortana p. 51 (2019)
8. Bode, G., Baranski, M., Schraven, M., Kümpel, A., Storek, T., Nürenberg, M., Müller, D., Rothe, A., Ziegeldorf, J.H., Fütterer, J., Scheuffele, B.: Cloud, wireless technology, internet of things: The next generation of building automation systems? J. Phys. Conf. Ser. **1343**(1) (2019)
9. Borrmann, A., König, M., Koch, C., Beetz, J.: Building Information Modeling: Why? What? How?, pp. 1–24. Springer International Publishing, Cham (2018)
10. Building Energy Data Exchange Specification (BEDES): Bedes (nd), accessed on October 17, 2024
11. BuildingSync: BuildingSync (nd), accessed on October 17, 2024
12. Cho, E.: Hey google, can i ask you something in private? In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. p. 1–9. CHI '19, Association for Computing Machinery, New York, NY, USA (2019)
13. Daniele, L., den Hartog, F., Roes, J.: Created in close interaction with the industry: The smart appliances reference (saref) ontology. In: Formal Ontologies Meet Industry. pp. 100–112. Springer International Publishing, Cham (2015)
14. Fitton, D., Read, J.C., Sim, G., Cassidy, B.: Co-designing voice user interfaces with teenagers in the context of smart homes. In: Proceedings of the 17th ACM Conference on Interaction Design and Children. p. 55–66. IDC '18, Association for Computing Machinery, New York, NY, USA (2018)

15. Følstad, A., Araujo, T., Law, E.L.C., Brandtzaeg, P.B., Papadopoulos, S., Reis, L., Baez, M., Laban, G., McAllister, P., Ischen, C., Wald, R., Catania, F., Meyer von Wolff, R., Hobert, S., Luger, E.: Future directions for chatbot research: an interdisciplinary research agenda. Computing **103**(12), 2915–2942 (2021)
16. Frens, J., Funk, M., van Hout, B., Le Blanc, J.: Designing the iot sandbox. In: Proceedings of the 2018 Designing Interactive Systems Conference. p. 341–354. DIS '18, Association for Computing Machinery, New York, NY, USA (2018)
17. Gokhale, G., Van Gompel, J., Claessens, B., Develder, C.: Transfer learning in transformer-based demand forecasting for home energy management system. In: Proceedings of the 10th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation. p. 458–462. BuildSys '23, Association for Computing Machinery, New York, NY, USA (2023)
18. Haller, A., Janowicz, K., Cox, S.J., Lefrançois, M., Taylor, K., Le Phuoc, D., Lieberman, J., García-Castro, R., Atkinson, R., Stadler, C.: The modular SSN ontology: A joint W3C and OGC standard specifying the semantics of sensors, observations, sampling, and actuation. Semant. Web **10**(1), 9–32 (2018)
19. Heck, M., Shon, S.H., Becker, C.: Does using voice authentication in multimodal systems correlate with increased speech interaction during non-critical routine tasks? In: 27th International Conference on Intelligent User Interfaces. p. 868–877. IUI '22, Association for Computing Machinery, New York, NY, USA (2022)
20. Huang, X., Zhang, S.: Human activity recognition based on transformer in smart home. In: Proceedings of the 2023 2nd Asia Conference on Algorithms, Computing and Machine Learning. p. 520–525. CACML '23, Association for Computing Machinery, New York, NY, USA (2023)
21. Janowicz, K., Rasmussen, M.H., Lefrançois, M., Schneider, G.F., Pauwels, P.: BOT: The building topology ontology of the W3C linked building data group. Semant. Web **12**(1), 143–161 (2020)
22. Jia, J., Ma, H., Zhang, Z.: Integration of industry foundation classes and ontology: Data, applications, modes, challenges, and opportunities. Buildings **14**(4) (2024)
23. Kim, D., Yoon, Y., Lee, J., Mago, P., Lee, K., Cho, H.: Design and implementation of smart buildings: A review of current research trend. Energies **15**, 4278 (06 2022)
24. Koshy, V., Park, J.S.S., Cheng, T.C., Karahalios, K.: "we just use what they give us": Understanding passenger user perspectives in smart homes. In: Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. CHI '21, Association for Computing Machinery, New York, NY, USA (2021)
25. Lamiae, E., Hicham, G.T., Fatiha, E., Mohammed, B., Hajoub, M.W.: Patient smart home monitoring using vision neural network transformers. In: Proceedings of the 6th International Conference on Networking, Intelligent Systems & Security. NISS '23, Association for Computing Machinery, New York, NY, USA (2023)
26. Lecorvé, G., Veyret, M., Brabant, Q., Rojas Barahona, L.M.: SPARQL-to-text question generation for knowledge-based conversational applications. In: Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on NLP (Volume 1: Long Papers). pp. 131–147 (Nov 2022)
27. Lee, T.H., Kim, H., Lee, D.: Transformer based early classification for real-time human activity recognition in smart homes. In: Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing. p. 410–417. SAC '23, Association for Computing Machinery, New York, NY, USA (2023)
28. Lin, C.Y.: ROUGE: A package for automatic evaluation of summaries. In: Text Summarization Branches Out. pp. 74–81. Association for Computational Linguistics, Barcelona, Spain (Jul 2004)

29. Louis, J., Rashid, K.: Utilizing building information models as operating systems for smart homes. In: Proceedings of the Workshop on Human-Habitat for Health (H3): Human-Habitat Multimodal Interaction for Promoting Health and Well-Being in the Internet of Things Era. H3 '18, Association for Computing Machinery, New York, NY, USA (2018)
30. Nguyen, H.D., Tran, T.V., Pham, X.T., Huynh, A.T., Do, N.V.: Ontology-based integration of knowledge base for building an intelligent searching chatbot. Sensors Mater. **33**(9), 3101–3123 (2021)
31. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL) pp. 311–318 (10 2002)
32. Project Haystack: Project haystack (2024), accessed on October 17, 2024
33. Sai, A.B., Mohankumar, A.K., Khapra, M.M.: A survey of evaluation metrics used for NLG systems. CoRR **abs/2008.12009** (2020)
34. Saini, A., Verma, A., Arora, A., Gupta, C.: Linguistic rule-based ontology-driven chatbot system. In: Bhatia, S.K., Tiwari, S., Mishra, K.K., Trivedi, M.C. (eds.) Advances in Computer Communication and Computational Sciences. pp. 47–57. Springer Singapore, Singapore (2019)
35. Salovaara, A., Bellucci, A., Vianello, A., Jacucci, G.: Programmable smart home toolkits should better address households' social needs. In: Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. CHI '21, Association for Computing Machinery, New York, NY, USA (2021)
36. Tatari, M., Agarwal, P., Alam, M.A., Ahmed, J.: Review of smart building management system. In: Tuba, M., Akashe, S., Joshi, A. (eds.) ICT Systems and Sustainability. pp. 167–176. Springer Nature Singapore, Singapore (2022)
37. Tawmo, T., Borah, M., Dadure, P., Pakray, P.: Comparative analysis of t5 model for abstractive text summarization on different datasets. In: Proceedings of the International Conference on Innovative Computing & Communication (ICICC) 2022 (April 29 2022)
38. Tran, H., Phan, L., Anibal, J., Nguyen, B.T., Nguyen, T.S.: Spbert: An efficient pre-training bert on sparql queries for question answering over knowledge graphs (2021)
39. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need (2023)
40. Vegesna, A., Jain, P., Porwal, D.: Ontology based Chatbot (For E-commerce Website). Int. J. Comput. Appl. **179**(14), 51–55 (2018)
41. Wang, M., Yang, N., Weng, N.: Securing a smart home with a transformer-based iot intrusion detection system. Electronics **12**(9) (2023)
42. Wang, R., Zhang, Z., Rossetto, L., Ruosch, F., Bernstein, A.: Nlqxform: A language model-based question to sparql transformer (2023)
43. World Wide Web Consortium (W3C): Semantic sensor network ontology.
44. Yao, Y., Huang, L., He, Y., Ma, Z., Xu, X., Mi, H.: Reviewing and Reflecting on Smart Home Research from the Human-Centered Perspective. Conf. Hum. Factors Comput. Syst. - Proc. (2023)