RESEARCH ARTICLE

Applied logic and semantics on indoor and urban adaptive design through knowledge graphs, reasoning and explainable argumentation

Evangelos A. Stathopoulos [ID], Alexandros Vassiliades [ID], Sotiris Diplaris , Stefanos Vrochidis and Ioannis Kompatsiaris

Information Technologies Institute, Center for Research and Technology Hellas, Thessaloniki, Greece
Corresponding author: Evangelos Stathopoulos; Email: estathop@iti.gr

Abstract

In the previous two decades, knowledge graphs (KGs) have evolved significantly, inspiring developers to build ever-more context-related KGs. Due to this development, artificial intelligence (AI) applications can now access open domain-specific information in a format that is both semantically rich and machine comprehensible. In this article, a framework that depicts functional design for indoor workspaces and urban adaptive design, in order to help architects, artists, and interior designers for the design and construction of an urban or indoor workspace, based on the emotions of human individuals, is introduced. For the creation of online adaptive environments, the framework may incorporate emotional, physiological, visual, and textual measures. Additionally, an information retrieval mechanism that extracts critical information from the framework in order to assist the architects, artists, and the interior designers is presented. The framework provides access to commonsense knowledge about the (re-)design of an urban area and an indoor workspace, by suggesting objects that need to be placed, and other modifications that can be applied to the location, in order to achieve positive emotions. The emotions referred reflect to the emotions experienced by an individual when being in the indoor or urban area, which are pointers for the functionality, the memorability, and the admiration of the location. The framework also performs semantic matching between entities from the web KG ConceptNet, using semantic knowledge from ConceptNet and WordNet, with the ones existing in the KG of the framework. The paper provides a set of predefined SPARQL templates that specifically handle the ontology upon which the knowledge retrieval system is based. The framework has an additional argumentation function that allows users to challenge the knowledge retrieval component findings. In the event that the user prevails in the reasoning, the framework will learn new knowledge.

1. Introduction

The knowledge representation of a cognitive robotic system can benefit from the use of semantic web knowledge bases (KBs) by gaining both explainability and scalability. Commonsense reasoning abilities can occasionally be found in semantic Web KBs such as ConceptNet (Speer *et al*., 2017) and WordNet (Fellbaum, 2010). An information retrieval architecture must also be able to defend and justify the decisions it provides (Young *et al*., 2016; Vassiliades *et al*., 2022).

Regardless of the complexity of the relevant entities in the query, an information retrieval system should ideally be able to provide answers to as many questions as possible; nevertheless, that is not always the case. This occurs because the knowledge that may be retrieved by any KB, which is chosen for information retrieval purposes is constrained in terms of quality for a number of different reasons. Using

external knowledge sources, more notably semantic Web KBs, which can offer commonsense information, is one potential option (Zamazal, 2020), to enhance the quantity of questions that an information retrieval system can answer.

An information retrieval framework should also provide justification for any responses it provides in response to a query. Argumentation is a popular way to support a position (Vassiliades *et al.*, 2021). An explanation that is understandable to humans must be provided by the information retrieval framework with the use of argumentation to support the accuracy of the answers it returns. If one's opinion is disproven through an argumentation discussion, argumentation can often serve as a technique of learning new information. Hence, argumentation can help an information retrieval system learn new information which can be utilized later.

Two distinct important ideas serve as the foundation for the framework, presented in this paper. The first is to enhance urban design in a city that is fast growing by addressing any new issues that may develop regarding its usefulness, attractiveness for mobility, and protection of culture and environment. Therefore, the framework will heighten sensitivity and understanding towards a city's present environmental and mobility challenges as well as its cultural value. The second, is helping to reorganize workspaces to achieve greater esthetics and utility. This increases opportunities for pleasant social contact or focused isolation in work environments, which improves productivity and creativity across departments and teams.

The main motivation of this paper is to provide a framework that could assist architect, artists, and interior designers to (re-)design an urban area or an indoor workspace, by adding objects or making modifications to the architecture of the location, so as afterward the location to provoke positive feelings to the individuals that act in the location. For this reason, in the KB of the framework several data are being mapped such as emotional, physiological, visual, and textual information which reflects emotionally relevant information of the indoor and urban location. Furthermore, users should be given access to a reasoning mechanism based on real-life circumstances in order for them to conveniently access the knowledge contained inside the framework. Experts in the relevant fields should ideally create the real-life scenarios.

The problem addressed in this paper is to provide a framework to the architects, artists, and interior designers, among others, which will assist them to have a complete view over the individuals' emotions, when (re-)designing a location. Therefore, helping in saving time and budget with unnecessary changes. The main aspects in which the framework can help the architects, artists, and interior designers, are the functionality, the memorability, and the admiration of the location, and the preserving of natural and cultural heritage.

For this reason, an information retrieval framework for emotions related to and functional design for the indoor and urban adaptive space design, enhanced with external knowledge sources, which can argue over the answers that it retrieves and learn new knowledge by means of an argumentation process, is proposed. The following research questions arise, for the framework presented in this paper: (1) 'How can a domain-specific information retrieval framework be extended, in order to retrieve knowledge that exists in multiple knowledge bases?', (2) 'How can an information retrieval framework support its answers in a way closer to the human way of reasoning?', (3) 'How can an information retrieval framework learn new knowledge with methods closer to human learning?', and (4) 'How can an information retrieval framework be evaluated?'. This study attempts to answer question (1) with its information retrieval component, questions (2) and (3) with its learning-through-argumentation component, and question (4) with its evaluation part, which presents various ways to evaluate a framework with user-driven evaluations.

The contribution of this paper is on one hand the KG which can represent multi-modal measurements, which in turn helps architects, artists and interior designers, among others, with the empirical and pragmatic perception of actual occupants, so as to drive the development of unconventional solutions in the design of spaces. On the other hand, another contribution is the information retrieval mechanism for the knowledge graph (KG), which proves to be helpful in real-life scenarios. Moreover, the aforementioned contributions can be classified into theoretical and empirical. Theoretical contributions include a) a multi-KB information retrieval methodology for the domain of indoor and urban adaptive design,

b) a semantic-matching algorithm that finds semantic similarity among entities of an internal KB with entities from external KBs, namely the KG of ConceptNet, using semantic knowledge from ConceptNet and WordNet, and c) a supervised method of learning-through-argumentation based on commonsense reasoning, rather than a data-driven model. The novelty of this work lies in the combination of these two cognitive processes (i.e., information retrieval and reasoning with argumentation for the information retrieved), which, to the best of our knowledge, has not been given much attention in the literature, in the context of indoor and urban adaptive design. However, it is important to have an information retrieval framework which can argue over the information that it responds with because it can convince the user about its answers with methods closer to the human way of thinking.

The framework is highly scaleable and able to represent data for any indoor workspace and metropolitan region. The framework is primarily designed for offices, but it may also be used in other indoor workspaces, such as at a cottage industry. Also, it might be a logical extension to include portraying feelings pertinent to outdoor workspaces and to advise individuals on how to improve the functionality, mobility, and appeal of an urban area with respect to the environmental and cultural relevance.

Notice that this study is the extension of Stathopoulos *et al*. (2023), extended by: (1) Enhancing the information retrieval mechanism of the KG with new real-life scenarios, (b) adding a semantic matching algorithm that uses external knowledge from ConceptNet and WordNet in order to help the framework answer questions about entities which do not belong in the KB of the framework, and (c) developing a learning-through-argumentation mechanism with which a user can argue with the framework over its answers, and if the argumentation dialogue concludes in favor of the user, it will result for the framework to learn new knowledge.

Empirical contributions of the framework highlights three points: (a) The completeness of the KG was evaluated with a user evaluation over the competency questions, where we asked from a group of individuals (20 in total), to answer the competency questions presented in Section 4 based on their own experience, and we compared their answers with the answers of the framework, and a matching of 78,5% was found, (b) the consistency of the KG was evaluated with a set of custom constraint rules created and enforced upon the KG where no violation warnings were detected, and (c) the high accuracy scores that the information retrieval mechanism achieved ($\sim$98%) when evaluated, shows that it can be used as an individual mechanism for helping users, by providing insightful information. Finally, the evaluation of the learning-through-argumentation component of the framework indicates that the component achieves its purpose to a large extent, as all argumentation scenarios got a satisfaction score close to 70%. Therefore, users find the argumentation dialogue rational and convincing.

The outline of this paper is the following. In Section 2, the related work of this paper is presented. Next, in Section 3, the nature of data, the KG, and the information retrieval mechanism of the framework is discussed. In Section 4, the learning-through-argumentation mechanism of the framework is showcased. Furthermore, in Section 5, how the KG, the information retrieval mechanism, and the learning-through-argumentation mechanism were evaluated is described. The paper concludes with Section 6 in which a detailed discussion and a conclusion with research directions for future potential extensions are given.

2. Related work

The most important components of this study are: the KG, the information retrieval mechanism which extracts knowledge from the KG, the learning-through-argumentation component, and the semantic similarity algorithm. For this reason, this paper separates the related work into four main subsections, one for similar KGs, one for the domain-specific information retrieval mechanisms for KGs, one for other cognitive mechanisms that use argumentation as a procedure of learning, and a last one which describes other similar semantic similarity algorithms.

Knowledge Graphs: The first category of KGs which might be considered close to the KG that we present in this paper is KGs for arts and artists. The area of KGs about arts and artists is not so

rich, as there are not many studies that could be classified clearly in this domain. One exception, is the study of Raven *et al.* (2020), where the authors present a KG that can represent the steps of specification, conceptualization, integration, implementation and evaluation in a case study concerning ceramic-glass. The difference with the previous KG is that our KG offers knowledge representation on how objects, landscapes, and buildings in an urban and objects in an indoor office environment, can be utilized in order to provoke specific feelings to the citizens/workers. KGs about cultural heritage could be argued that are close to the utilization of landscapes our KG does, but cultural heritage KGs such as Hyvönen (2012), Schneider (2020), usually focus on knowledge for preservance of the landscapes, while we focus on how the landscapes can be utilized in order to provoke positive feelings to the citizens.

Therein after, the area of KGs for architecture and architects seems to be quite richer. Lopes (2007), showcases the main notions that must be represented in a KG about architectural concepts. Additionally, KGs like Wagner and R̈uppel (2019), Kumar *et al.* (2019), which are considered to belong in the area of architecture, mostly contain information about materials of objects and their uses. For the former, it consists a theoretical study, while this paper offers a constructed fully-fledged KG. For the latter two KGs, the KG presented in this paper differs in regards of information that can represent. More specifically, the KG in this paper represents knowledge on how objects, landscapes, and buildings in an urban and objects in an indoor office environment, can be utilized in order to provoke specific feelings to the citizens/worker, while the previous studies mostly contain information about materials and dimensions of objects.

Information Retrieval: The area of information retrieval over KGs is quite rich and was enhanced in the last decade with the constant evolution of KGs. The reason why KGs are very helpful in retrieving information, is because they have a predefined format, which is easily understandable by machines, and their terminology and assertion components can be defined based on a set of rules which can represent commonsense knowledge. We referred previously to the term domain specific information retrieval, which is different than simple information retrieval. As domain specific information retrieval contains the domain specific knowledge that contain the details required, for a better representation of knowledge for a domain. For example, a generic rule that defines the distribution of household objects across any domain works, but it works better if we know that in the household domain it is natural to see coffee next to sugar and kitchen utensils, while in the domain of a store the same objects can be distributed per aisle according to their type.

Even though many general techniques have been presented (Munir & Anjum, 2018; Asim *et al.*, 2019; Yu, 2019), a lack of studies is imminent for information retrieval over KGs restricted to a specific domain. The reason why it is considered hard to create an information retrieval mechanism over KGs which are restricted to a specific domain, is because domain experts are needed in order to create real-life scenarios, based on which the information retrieval mechanism should be constructed (as mentioned in Chi *et al.*, 2019). Vassiliades *et al.* (2020, 2022) are an exception as a information retrieval mechanism for the household environment is presented, but in contrast to the study the scenarios presented there, are not a result of knowledge provided by domain experts.

Semantic Similarity Algorithm: The semantic matching algorithm was mostly inspired by the works of Young *et al.* (2016), and Icarte *et al.* (2017) where they use CS knowledge from the web ontologies DBpedia, ConceptNet, and WordNet to find the label of unknown objects. As well as from the studies (Young *et al.*, 2017; Chernova *et al.*, 2020), where the label of the room can be understood through the objects that the cognitive robotic system perceived from its vision module. One drawback that might be noticed in these works is that all of them depend on only one ontology. Young *et al.* compares only the DBpedia comment boxes between the entities, Icarte et. al. acquires only the property values from ConceptNet of the entities, and Chernova *et al.* (2020), Young *et al.* (2017) on the synonyms, hypernyms, and hyponyms of WordNet entities.

Learning-through-Argumentation: The idea of learning through an argumentation dialogue is an innovative idea which has not been given much attention in the field of machine learning. It is a very interesting method to allow an individual (i.e., machine or human) to learn new knowledge, because it resembles a commonsense reasoning procedure which is closer to the human way of thinking. This

method is applied in schools (Von Aufschnaiter *et al*., 2008; Berland & McNeill, 2010; Akhdinirwanto *et al*., 2020; Fiorini, 2020; Hu *et al*., 2022), to allow students to argue and learn through argumentation by accepting other students' arguments, if they cannot defend successfully their position. In most cases the need of a supervisor, a teacher in most cases which will check the rationality of the arguments, is mandatory. This method of learning is also found in collaborative learning (Veerman, 2000), between a human and a machine at a theoretical level. However, some data driven models that can be trained over datasets of argumentation dialogues for legal cases have been presented in Možina *et al*. (2005, 2007). The difference is that the framework presented in this paper does not need training in order to perform argumentation dialogues, and it can learn through the dialogue.

Collaborative learning through argumentation in multi-agent systems has been proposed in Ontañón and Plaza (2007), Ontañón and Plaza (2010). The authors of these papers present some protocols upon which a group of agents can start learning from each other to improve the individual and joint performance for decision making. Moreover, the authors state that 'argumentation is a useful framework for joint deliberation and can improve over other typical methods such as voting'. The difference is that these are theoretical studies which propose protocols that agents should follow, to enhance learning-through-argumentation. Also, they solve conflicts through preference rules which needs tuning based on the context of the conversation. Instead, the framework presented in this paper uses commonsense knowledge from human individuals to learn new knowledge. Learning-through-argumentation is also presented in Chen *et al*. (2019), Drapeau *et al*. (2016), and Clark *et al*. (2007), Slonim *et al*. (2021). In Chen *et al*. (2019), Drapeau *et al*. (2016), the authors present a framework that human individuals can learn from each other, and in Clark *et al*. (2007), Slonim *et al*. (2021) external knowledge from the Web is used for humans to learn. Instead, this paper focuses on agent learning.

Another way to view this part of the framework, is that a method of knowledge refinement through argumentation is introduced, that supervises the knowledge populating the KB by using the commonsense of the user. One can notice that belief revision is a method of learning (Kelly, 1998); an agent removes or adds knowledge to its KB to learn. Therefore, argumentation can be considered as a method of belief revision (Vassiliades *et al*., 2021, 2022), because argumentation can convince the opposing participant(s) to refine the knowledge in their knowledge base. Belief revision with argumentation has been used for various tasks such as to enhance the knowledge in a KB (Rahwan *et al*., 2005; Cayrol *et al*., 2008; Falappa *et al*., 2009), to explain why an argument is true or false (Coste-Marquis *et al*., 2014a,b; Fan & Toni, 2015), and for negotiation (Okuno & Takahashi, 2009; Pilotti *et al*., 2014, 2012). But to the best of our knowledge, studies about knowledge refinement through an argumentation procedure, remain at a theoretical level.

Learning through case-based argumentation (Aleven & Ashley, 1997; Ashley *et al*., 2002; Fan *et al*., 2013; Heras *et al*., 2013; Čyras *et al*., 2016), is a method for learning how to classify an argument. In case-based argumentation the argument is classified as acceptable, or non-acceptable based on similar examples of argument classification. Therefore, arguments are classified using similarity metrics extracted from a KG, or based on feature importance, or employing embeddings, among others. Instead, in this work the user's commonsense is employed to classify an argument as acceptable, or not.

3. The knowledge representation of the framework

The KG is part of a more complex service-oriented architecture. Therefore, the KG communicates with other services and components. This is mentioned because the visual, textual, and stress level analysis components will not be described in detail. Each one of the aforementioned components produces a message, in json format, which is passed to the mapping mechanism of the KG. Subsequently, the mapping mechanism will parse the information in each json, and will map the most crucial information in the KG. The source code of the mapping mechanism can be found here[1]. The KG is populated with some messages from the visual, textual, and stress level analysis components, then a user can hand-pick from a set of predefined SPARQL queries, in order to access the information inside the KG. Once the user
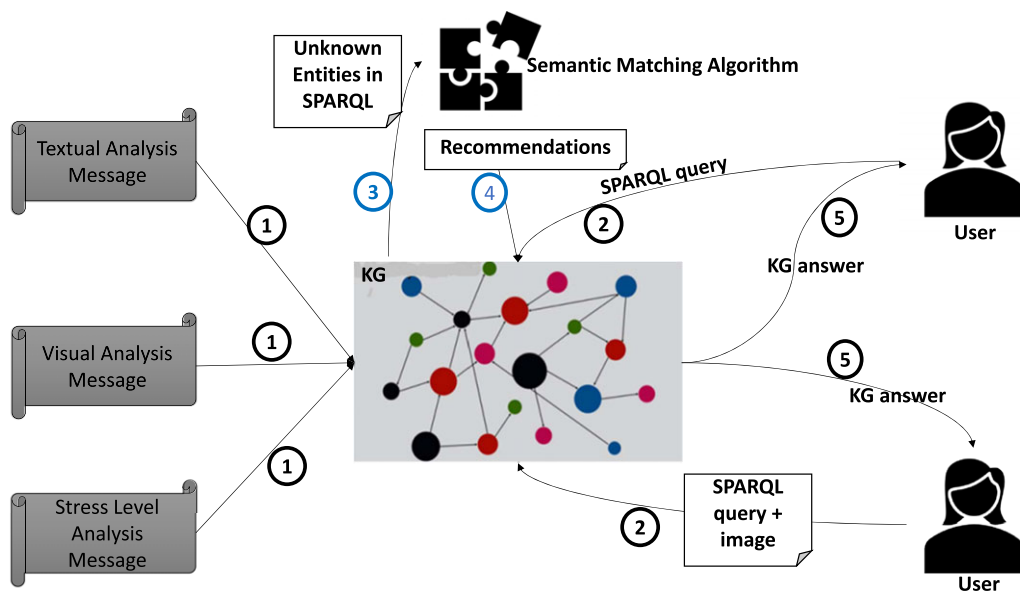
---

[1]https://github.com/valexande/MindSpacesPUC1-2

Figure 1. *Pipeline of the KG*

chooses a specific query, the information retrieval mechanism of the KG is triggered and the information is returned to the user. Notice that all SPARQL queries were defined with the help of domain experts, and some of them also require an image as an input in order to cast the SPARQL query. Additionally, if the user asks a query which contains entities that do not exist in the KG, the Semantic Matching Algorithm (see Section 3.4) will take action, and will provide a recommendation from external knowledge sources that is similar with the entity that was in the user's query.

Figure 1, shows an outline of the pipeline, each number in the circles shows the order of steps. Notice that the blue circles mean that the step will not mandatorily be executed, but only if the conditions are valid, in this case if the user's query contains an entity that does not exists in the KG.

3.1 Nature of data

The multi-modality and variety of data flowing in the system and the necessity of homogenization and fusion mandated the adoption of a semantic KG to address likewise requirements. The KG is not responsible for archiving and storing raw data files, since there is an underlying data storage facility for that purpose. Instead, the KG hosts metadata of raw data, analyses results and miscellaneous information with semantic value among other candidates for being mapped and fused into the knowledge base. One can find a blueprint of the messages fused in the KG here[1]. The KG is accompanied with novel ontological models to achieve proper semantic annotation of the raw data.

The main categories of data which needed to be captured in the KG were: physiological analysis results from galvanic skin responses (GSR), visual analysis results from images, textual analysis results derived from online retrieved content, and general information about VR experiments. Some of the aforementioned analyses results share some common properties such as in the case of the imageability metric.

The physiological signals (GSR) were captured during hot-spot or navigation VR experiments where multiple users conducted stress-induced tasks inside virtual environments containing multiple configurations of work offices. The ultimate goal was to obtain and assess stress indicators directly from the skin of the subjects while experiencing different setups of the space.

The visual analysis component consists of several machine learning models either trained from scratch or fine-tuned from other efforts. The model performing semantic segmentation (Qiu *et al.*, 2021)

on images was trained to extract semantic labels and percentages per pixel on images while the Verge classifier (Andreadis *et al.*, 2020) was deployed to classify the images to one or more classes based on context. The valence-arousal model for exterior design was trained on a newly collected dataset annotated by experts to deliver confidence and values on a happy-unhappy and calm-excited scale, while a visual imageability score is generated (Pistola *et al.*, 2022).

Next, we describe the messages that are received from the visual, textual, and stress level analysis components and are passed in the KG. As mentioned these messages come in the form of json files and are mapped into the ontology. We start with the visual analysis messages, where a high level analysis of the json file is that it contains information about the segmented objects along with their percentages in the image, the segmented objects colourfulness (i.e., how bright their colours are in an image), the verge classification (i.e., the label given to the content of the image; for a example an image can be given the label park), their probabilities (i.e., confidence of the classification), and the valence and arousal with their probabilities (i.e., the probabilities to provoke these two feelings). Also, some metadata such as the timestamp are included in the json file. One can see a visual analysis message below, notice that the number of objects in the message were reduced for beautification reasons.

```
                     Visual Analysis Message
{
  "timestamp": "2022-03-17 10:36:22.541764+01:00CET",
  "segmentation": {
    "segm_labels": [
      "tree",
      "wall"
    ],
    "segm_percentages": [
      0.2822998046875,

      0.0000244140625
    ],
    "segm_colorfulness": [
      22.635735826237845,
      35.04844901850117
    ]
  },
  "classification": {
    "verge_labels": [
      "/j/jewelry_shop"
    ],
    "verge_probabilities": [
      0.38903313875198364
    ]
  },
  "sentiment": {
    "valence": "positive",
    "valence_probability": 0.5103972554206848,
    "arousal": "excited",
    "arousal_probability": 0.6642298698425293
  }
}
```

The textual analysis message contains information about the textual description of the image, and the emotion that caused to the user. Moreover, it contains information about the named entities found in the textual description (i.e., the verbs and nouns in the sentence), their type (i.e., if they are a concept or not), their label, their DBpedia link (if a historical sight or a landmark is found), the begin and end timestamp (i.e., the moment they are spotted). The textual analysis message also contains some metadata information for the named entities and the image, which are the contributions of the named entities in the imageability score of the image, the overall imageability, colour imageability, and segmented imageability. Imageability is a value that measures how memorable a image is to a user, defined by architects which takes into contribution the colourfulness and segmentation of an object if we refer to an object, and of all the objects contained in an image if we refer to the imageability of an image.

```
                    Textual Analysis Message
{
  "id": "JPEG_20220204_153627_4494048047500552089.jpg",
  "text": "Un lloc on quedava i on quedo amb amics i companys.",
  "language": "ca",
  "result": {
    "ner": {"directed": true, "type": "ner", "label": "ner",
    "emotion_detection": { "directed": true, "type": "sentiment",
                            "label": "arousal"
    },
    "imageability_scoring": {
      "label": "imageability scoring",
      "imageability": 5.346666666666667,
      "covered_text": "Un lloc on quedava i on quedo
                        amb amics i companys.",
      "concept_wise_imageability": [ 4.33, 5.62 ],
      "concept": [ ["tree"],["wall"]],
      "final_contribution": [4.33, 5.62],
      "colourfulness": [ 22.635735826237845, 35.04844901850117],
      "segmentation": [0.3939639627933502,0.33205193281173706]
    },
    "verge": {
      "label": [["street"], ["residential"],["neighborhood"]],
      "verge_percentage": [0.3502,0.3320573706,0.3706]
    },
    "textual_metadata": {
      "date": "2022/03/10 13:38:51",
      "PARSING": "13:38:45",
      "EMOTION": "13:38:45",
      "NER": "13:38:46",
      "CONCEPT_CANDIDATES": "13:38:46",
      "DBPEDIA": "null",
      "IMAGEABILITY": "13:38:51"
    }
  }
}
```

The multimodal mapping mechanism can also map into the KG, the stress level the individuals experience according to their location in a workspace. These data are collected through stress level analysis sensors located on individuals. By letting them to actively browse in their workspace, in a regular work day. The stress level analysis json contains the location of the user in a workspace, the stress level (s)he experiences, the focus for work, and the privacy one has in each location. The location of the user even tough is given in x,y,z coordinates, for each workspace different areas (i.e., rooms and other workspace location such as the parking) would be defined by a clustering function. The clustering function is a geometrical clustering of x,y,z coordinates, where if many x,y,z coordinates would be found geometrically close, an area would be defined. In the example below 3 such areas exist.

```
                 Stress Level Analysis Message
{
  "Zone_1": { "focusWork": 5, "privacy": 4.2, "stress": 1.3,
    "location": [
      {"x": 13.299571, "y": 0.0, "z": 5.93938732},
       .                ...
      {"x": 13.2288609, "y": 0.0, "z": 5.868676}
    ]
  },
  "Zone_2": {"focusWork": 4.1, "privacy": 4.1,"stress": 2.2,
    "location": [
      {"x": 1.990005, "y": 0,"z": 5.105028},
       .                ...
      {"x": -5.476165, "y": 0, "z": 6.087821}
    ]
  },
  "Zone_3": {"focusWork": 2.8,"privacy": 2.5,"stress": 2.6,
    "location": [
      {"x": 4.14202642, "y": 0, "z": 6.358342},
       .                ...
      {"x": 4.08192635,"y": 0,"z": 6.41263676}
    ]
  },
  "user_id": 11
}
```

3.2 The KG

The KG is separated into two big sub graphs. One for the purpose of representing knowledge that aims at improving urban design in a rapidly expanding city by addressing new challenges that may arise related to its functionality, mobility, attractiveness, protection of culture and environment. The other aims at representing knowledge that involves increasing opportunities for positive social interaction in work environments which leads to improved productivity and creativity across departments and teams, by helping to readjust workspaces to achieve better aesthetics and functionality.

The subgraph for the first case was constructed based on the information received from the various components. In more detail, the messages which come from the textual and visual analysis components

were analysed (see sub Section 3.1), and a KG was developed based on those. In the corresponding part of the ontology the important concepts of these messages were represented through classes and relations among them. Moreover, the requirements of the information retrieval mechanism were analysed, meaning that the competency questions given by users and experts were taken into consideration, and we defined classes and relations in such a way that would help the information retrieval mechanism retrieve the crucial information easily. Notice that a competency question, is a question, which a user would like to be answered by the KG. In Figure 2 one can see the classes and the object type properties of the first subgraph.

The namespace *mind1* is used to indicate the classes and relations for the first subgraph of the KG. Next, a detailed analysis of the classes and the relations between them is given.

- The Sentence class contains information about the sentences that compose a textual description of an image (contained in the textual analysis json; see sub Section 3.1).
- The Text class contains information about the textual description of an image. The Text class is connected through the property hasSentence with the class Sentence, in order to give further information about the sentences that compose the textual description. Moreover, it has information for the language of the textual description, and the textual description itself. Finally, the Text class is connected through the property hasNamedEntity with the NamedEntity class, which has information about named entities found in the textual description. Named entities can be words that refer to real life objects, actions, or activities.
- The Named Entity class contains information about the named entities found in the textual description of an image (mentioned as concepts in the textual description of the image; see textual description message in sub Section 3.1). The NamedEntity class gives information for the category of the named entity relation (the category of a named entity is a classification that was given by the domain experts), the imageability score of the named entity, and is connected through the property hasURI with the class URI. The imageability score is a confidence score that is composed by the visual analysis component. Imageability according to the urban planner Kevin Lynch is the quality of a physical object to evoke a strong image in any observer, thus being memorable (Lynch *et al*., 1960).
- The URI class contains information about the URIs of the named entities. Currently, the URIs point only to DBpedia entities. The URI class has information about the URI link and the confidence that a named entity should be related with a specific URI.
- The VergeLabel class contains information about the labels of the verge classifications found in an image. Verge classification is the classification of the scenery that the image represents, for example a verge label can be a park. Notice that an image may have more than one verge classifications.
- The VergeContainer class contains information about the imageability scores of the segmented objects in the image that gave the verge classifications.
- The SemSegLabel class contains information about the labels of the segmented objects found in an image.
- The SemSegContainer class contains information about the segmented objects found in an image, for example the percentage of space they capture in the image, and the confidence that they are part of the image. Moreover, it is connected though the property hasSemSegLabel with the class SemSegLabel, in order to indicate the label of a segmented object.
- The Arousal class contains information about the arousal score that was given by a user for an image. Arousal is a confidence score given by the users.
- The Valence class contains information about the valence score that was given by the user for an image. Valence is a confidence score given by the users.
- The Image class is the most important class, as it contains a lot of metadata information about the characteristics of the image, such as the latitude, longitude, the pitch, the zoom, and others. But apart from these it is connected: (i) with the Arousal class through the property hasArousal
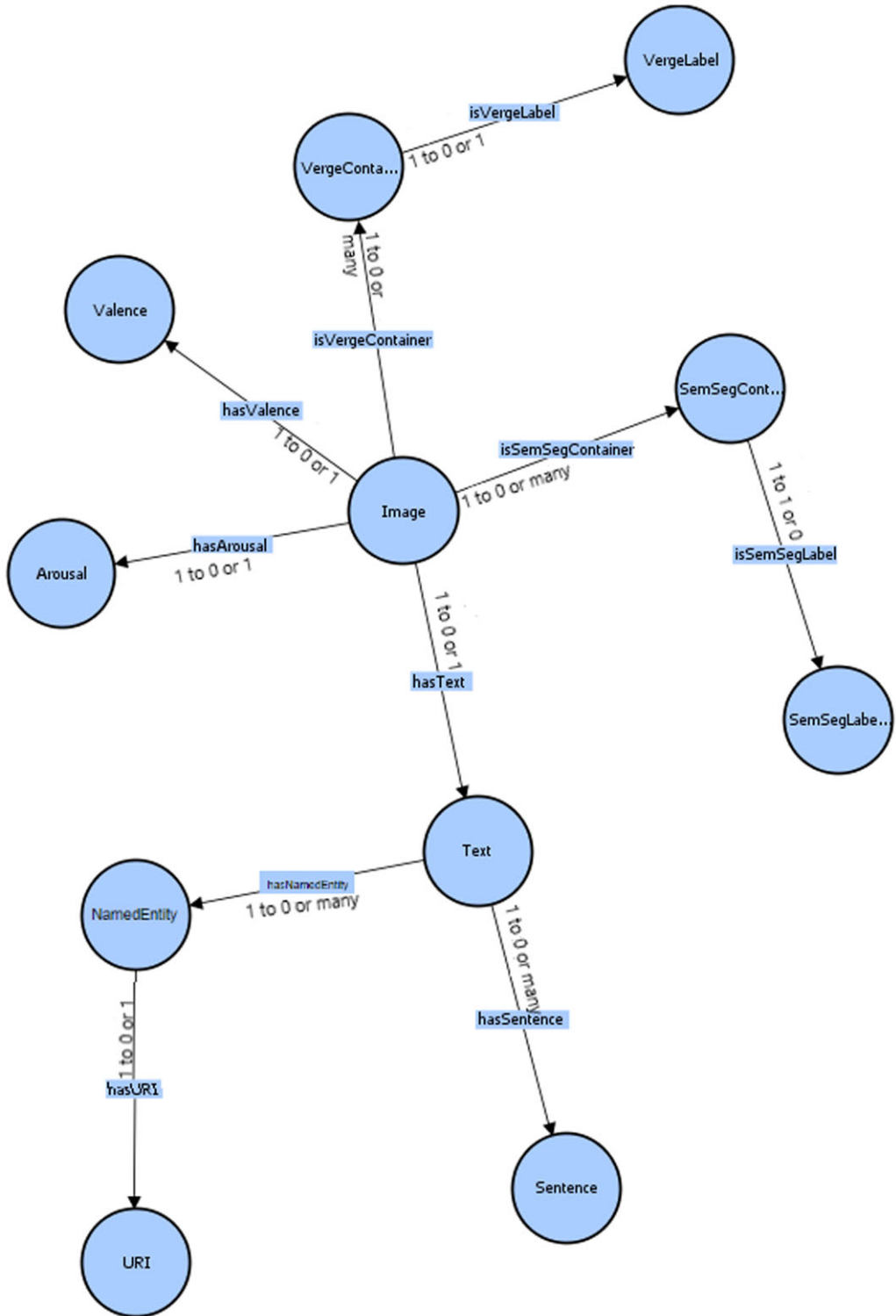
Figure 2. *The schema of the first subgraph of the KG*
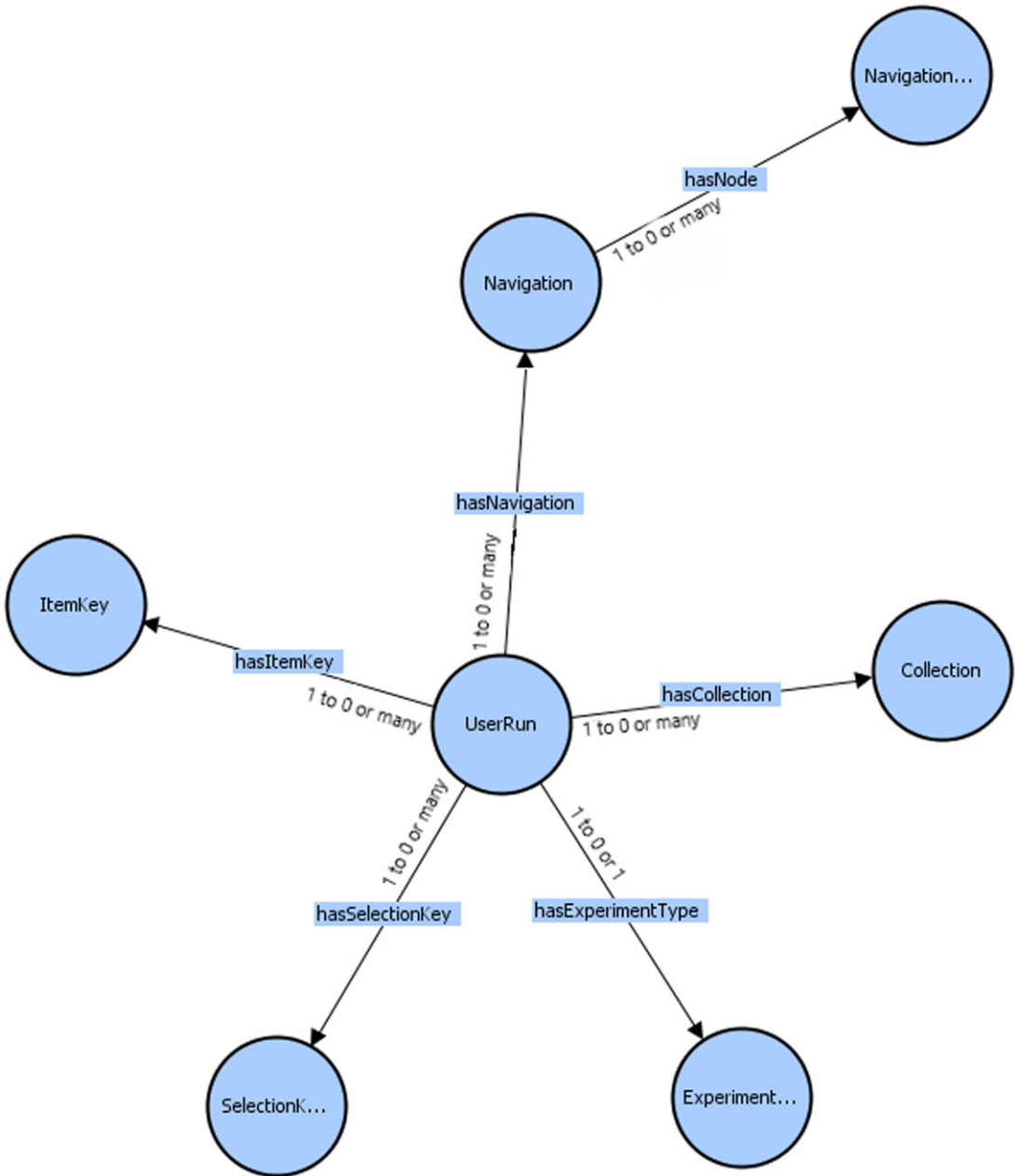
Figure 3. *The schema of the second subgraph of the KG*

to indicate its arousal, (ii) with the Valence class through the property hasValence to indicate its valence, (iii) with the VergeContainer class through the property isVergeContainer to give information about the verge classifications that it has, (iv) with the SemSegContainer class through the property isSemSegContainer to give information about the segmented objects it contains, and (v) with the Text class through the property hasText to give information about the textual description that it has.

In Figure 3 one can see the classes and the object type properties of the second subgraph.

The namespace *mind2* is used for the second subgraph. Next, follows a detailed analysis of the classes and the relations of the second subgraph.

- The Collection class contains information about the experiment status, meaning that it contains information about when the stress level sensors have started counting and when they stopped.
- The ExperimentType class contains information about the task that the user is performing at some given moment, for instance the user may be working, navigating in the workspace, having lunch, being in a meeting, or take a break, among others.
- The Navigation class contains information about the navigation that the user performed during a sensor measurement, for instance (s)he went from point A to point B. The navigation measures are counted in x, y, z coordinates. Moreover, it is related through the property hasNode with the class NavigationNode that contains information about the navigation instances (i.e., navigation nodes).
- The NavigationNode class contains the x, y, z nodes in which measurements have been taken, measurements such as the collaboration level between individuals, focus work, privacy, and stress level.
- The ItemKey class contains information about the measurements given by the users, the collaboration, focus work, privacy, and stress scores. The purpose of these scores, is to compare the scores in stress that the users have.
- Finally, the UserRun class is the most important class, as it connects the information from the aforementioned classes with a user run. More specifically the UserRun class is connected: (i) with the ExperimentType class through the property hasExperimentType to indicate the experiment types that it contains, (ii) with the ItemKey class through the property hasItemKey to indicate the item keys that it contains, (iii) with the Collection class through the property hasCollection to give information about the collections that it contains, and (iv) with the Navigation class through the property hasNavigation to give information for navigation nodes that it contains.

Notice that the two subgraphs eventhough they refer to different notions, i.e., the first subgraph is for the outdoor adaptive design and the second for the redesign of internal workspaces, they are related between them with the property hasImage. The property hasImage has domain the class Collection and range the class Image. The reason for that is because a collection is a set of images.

3.3 Information retrieval

The information retrieval mechanisms take advantage of the KG which was created to support the use cases of urban environments and indoor workspaces, as well as of the population of the knowledge base with content and metadata deriving from both artists and users.

The main idea, for the information retrieval over the first subgraph (see Subsection 3.2), was to feed on demand an interactive google map with geolocated 2D points in the form of CSV files corresponding to image entries inside the knowledge base through application programming interfaces (APIs) developed in node.js and Javascript. The graphdh.js library was used to establish the connection and transactions towards and from the GraphDB repository[2] with authorization and authentication enforced. The file delivered in response follows a scalable and dynamic approach, meaning it is being generated on demand based on live requests, thus ensuring always up-to-date data delivery as the knowledge base supports a continuous online population and new entries may arrive anytime. The format of the file consists of as many lines as the images fulfil the SPARQL queries and 5 columns:

- latitude (the latitude when the corresponding image was captured)

---

[2]http://160.40.52.169:6161

- longitude (the longitude when the corresponding image was captured)
- point size (the size of the circle to be depicted on the map)
- point opacity (the opacity of the circle to be depicted on the map)
- point colour (the colour of the circle to be depicted on the map)

In total, 16 SPARQL queries were formulated, each satisfying a different user requirement, followed by an additional multipurpose sparse function. Sparse function's purpose is to diminish the resulting pool of points and provide some sort of information aggregation. More specifically, it rounds the latitude, the longitude and the distance. Then, it clusters points first by latitude, then by longitude and finally by direction. Afterwards, it merges on imageability scores, segmentation labels and averages the percentages. Moreover, it merges on verge labels and averages their percentages. Finally, the function merges on sentiment data.. For the first use case (i.e., that of the outdoor adaptive design) the following 6 SPARQL queries were formulated. The purpose of the following 6 SPARQL queries is to help architects compare which objects, landscapes, and buildings contain the images which provoke specific feelings to citizens. For instance, if the image which contains object X, Y, Z provoke the feeling of arousal to the citizen, these objects might be important to be included in a scenery in order to provoke similar feelings. The feelings that were formulated are arousal, valence, meaning that a picture might provoke (or not) arousal and/or valence to a citizen, therefore some objects, landscapes and buildings might have negative impact to the feelings of the citizen.

Query 1: Retrieve images ordered by all image entries having latitude, longitude, direction, segmentation imageability, verge imageability, valence with value and probability, arousal with value and probability, all segmentation labels with percentages and all verge labels with percentages. The point size is set to 25 and the point opacity is set to 0.03. All points are given the colour red.

Query 2: Retrieve images ordered by all image entries having latitude, longitude, direction, segmentation imageability, verge imageability, valence with value and probability, arousal with value and probability, all segmentation labels with percentages, all verge labels with percentages and the landmark, node or edges statuses optionally if they exist. If the point is a landmark set colour to green, if it is an edge set colour to red, else if the point is a node set colour to blue. The point size is set to 33 and the point opacity to 1.

Query 3: Retrieve images ordered by all image entries having latitude, longitude, direction, segmentation imageability, verge imageability, valence with value and probability, arousal with value and probability, all segmentation labels with percentages and all verge labels with percentages. If valence is 'positive', set the green channel to 88 HTML colour code. If valence is 'neutral', set red, green and blue channels to 88 HTML colour code. If valence is 'negative', set the red channel to 88 HTML colour code. If arousal is 'calm' and if the green channel is unset, set the green channel to 88 HTML colour code, else set the green channel to FF HTML colour code. If arousal is "neutral" and if the red channel is unset, set the red channel to 88 HTML colour code else set it to FF HTML colour code. If the arousal is "neutral" and and if the green channel is unset, set it to 88 HTML colour code, else set it to FF HTML colour code. If the arousal is 'neutral' and the blue channel is unset, set it to 88 HTML colour code, else set it to FF colour code. If the arousal is 'excited' and the red channel is unset, set it to 88 HTML colour code, else set it to FF HTML colour code. Finally, concatenate the red, green and blue resulting HTML colour channels in a unique string to decide the final colour of the point. Point size is set to 30 and point opacity is set to 0.01

Query 4: Retrieve images ordered by all image entries having latitude, longitude, direction, segmentation imageability, verge imageability, valence with value and probability, arousal with value and probability, all segmentation labels with percentages and all verge labels with percentages. Check the top 1 segmentation class and if it is a building or vegetation or open sky, give the point the colour red, green or blue, respectively. The point size is set to 25 and the point opacity is set to 0.3.

Query 5: Retrieve images ordered by all image entries having latitude, longitude, direction, segmentation imageability, verge imageability, valence with value and probability, arousal with value and probability, all segmentation labels with percentages and all verge labels with percentages. Then group

by green colour 'nature and open spaces' if the top verge label is: zen garden, park, picnic area, botanical garden, orchard, broadleaf forest, playground, farm, cultivated field, vineyard, pasture, promenade, plaza, pavilion, exterior balcony, or amphitheater. Then group by blue colour 'built urban spaces' if the top verge label is: residential neighbourhood, outdoor apartment building, downtown, cemetery, outdoor hotel, or hospital. Finally, group by red colour 'disused or avoided spaces' if the top verge label is tundra, parking lot, driveway, wild field, landfill or sandbox. For all points, point size is set to 25 and point opacity to 0.3.

Query 6: Retrieve images ordered by all image entries having latitude, longitude, direction, segmentation imageability, verge imageability, valence with value and probability, arousal with value and probability, all segmentation labels with percentages, all verge labels with percentages and optionally the total imageability score. Then by using a rainbow colour function give a rainbow colour to the point scaling, respectively, from 0 to 7 scores.

For the second use case, i.e., that of the emotions related to the indoor workspaces the following SPARQL queries were formulated.

Query 1: Retrieve the average user's stress for each room in the workspace.

Query 2: Retrieve the total time needed to navigate in each room of the workspace by every user.

Query 3: Retrieve the overall design ratings for each room in the workspace.

Query 4: Retrieve the overall design ratings for each room in the workspace, based on the type of the user (i.e., if (s)he is a manager, a security worker, a driver, or an desk office worker according to the categorization results of a questionnaire[3] which was filled by the users prior to the data capturing).

Query 5: Retrieve how many users of each profile exist (i.e., if (s)he is a manager, a security worker, a driver, or an desk office worker).

Query 6: Retrieve which rooms in the workspace were classified by all users as most/least favourite based on the focus work, the privacy, and the collaboration.

Query 7: Retrieve which rooms in the workspace were classified by each user's type (i.e., pioneer, security worker, driver, or desk office worker) as most/least favourite based on the focus work, the privacy, and the collaboration.

Query 8: Retrieve the users stress over all rooms in the workspace.

Finally, two more SPARQL queries will be analysed named Scenario A and Scenario B, which were defined by domain experts as real-life scenarios.

Scenario A: Given a snap image, return a list of images, where: (i) the Top 3 segmentation labels of the snap (based on the coverage percentage) exist in the images, (ii) the images must have imageability $\geq$ imageability of snap $+ 0.05$, (iii) the Top 3 segmentation labels must exist in the images with the same coverage percentage, or a 20% difference, and (iv) the results must be limited to 8 images, if there exist as many, sorted by their imageability.

Scenario B: Given a snap image, return a list of images, where: (i) the Top 3 segmentation labels of the snap (based on the colourfulness percentage) exist in the images, and their colourfulness is above 1, and (ii) for each one of the 3 segmentation classes bring the Top 5 images with the highest colourfulness for each segmentation classes.

Notice that the queries which are refered in this subsection are also Competency Questions (CQs), except from **Scenario A** and **Scenario B** which are a combination of CQs. Notice that Scenario A and Scenario B, are use cases indicated by domain experts, which represent real-life scenarios for which the information retrieval should be capable to return information.

We present below **Query 1** of the indoor workspace use case, notice that in the example below we had 4 rooms.

---

[3]https://www2.deloitte.com/content/dam/Deloitte/us/Documents/process-and-operations/us-business-chemistry-infographic.pdf.

```
                Query 1 of Indoor Workspace Use Case

SELECT * WHERE {
    SELECT (SUM(?gsr1)/COUNT(?gsr1) as ?pr1)
    (SUM(?gsr2)/COUNT(?gsr2) as ?pr2)
    (SUM(?gsr3)/COUNT(?gsr3) as ?pr3)
    (SUM(?gsr4)/COUNT(?gsr4) as ?pr4) WHERE{
    ?s rdf:type :UserRun.
    ?s :hasNavigation ?nav.
    ?nav :title ?t.
    {filter contains(?t,"NavTest1") {
    ?nav :hasNode ?node.
                ?node :gsr ?gsr1}}
    UNION{
    filter contains(?t,"NavTest2") {
    ?nav :hasNode ?node.
            ?node :gsr ?gsr2}
        }
    UNION{
    filter contains(?t,"NavTest3") {
    ?nav :hasNode ?node.
            ?node :gsr ?gsr3}
        }
    UNION{
    filter contains(?t,"NavTest4") {
    ?nav :hasNode ?node.
            ?node :gsr ?gsr4}
        }
    }

}
```

3.4 Semantic matching algorithm

Since the dataset, upon which the knowledge retrieval framework was constructed, has a specific number of objects, in order to retrieve knowledge about objects on a larger scale, a mechanism that can take advantage of the web KGs DBpedia, ConceptNet, and WordNet to answer queries about objects that do not exist in the KB of framework was developed.

This would broaden the range of queries that the framework can answer, and would overcome the downside of the framework being dataset oriented. The user with the Semantic Matching Algorithm (SMA) can address queries with labels that do not exist in the internal KB of the framework. An aspect which could not be achieved without the SMA.

Algorithm 1 was implemented using Python. The libraries Request and NLTK[4] offer web APIs for all three aforementioned ontologies. Similar methods can be found in Icarte et al. (2017), Young *et al.*

---

[4]https://www.nltk.org/

([2016](#)), where they also exploit the commonsense knowledge existing in web ontologies. Algorithm 1 starts by getting as input any word that is part of the English language; this is checked by obtaining the WordNet entity, line 1. Note that any entity that is part of WordNet is also part of ConceptNet. The input is given by the user indirectly, when (s)he gives a keyword in a query that does not exist in the KB of the framework.

Subsequently, the algorithm turns to ConceptNet, and collects the properties and values for the input word, line 2. In the framework, only the values of the properties RelatedTo, UsedFor, AtLocation, CapableOf, Causes, ReceivesAction, and IsA are collected. These properties were chosen because they are the most related to the target application of providing information for the indoor and urban adaptive design. Also, the weights that ConceptNet offers for each triplet are acquired. These weights represent how strong the connection is between two different entities with respect to a property in the ConceptNet graph, and are defined by the ConceptNet community. Therefore, a hash map of the following form is constructed:

$$\{Property_1 : [(entity_1^1, (weight_1^1), \dots$$

$$(entity_m^1, (weight_m^1)], \dots$$

$$Property_l : [(entity_1^l, (weight_1^l), \dots$$

$$(entity_n^l, (weight_n^l)]\}$$

for $m, l, k \in \mathcal{N}^*$.

Then, the semantic similarity between the given entity and the returned property values is extracted using WordNet and DBpedia, lines 3–6. First, the algorithm finds the least common path that the given entity has with each returned value from ConceptNet, in WordNet, line 7. The knowledge in WordNet is in the form of a directed acyclic graph with hyponyms and hypernyms. Thus, in each case, the number of steps that are needed to traverse the path from one entity to another is obtained. Subsequently, the algorithm turns to DBpedia to extract comment boxes of each entity using SPARQL, lines 9–11. If DBpedia does not return any results, the entity is searched in Wikipedia, which has a better search engine, and with the returned URL DBpedia is asked again for the comment box, based on the mapping scheme between Wikipedia URLs and DBpedia URIs, lines 12–18. Notice that when a redirection list is encountered the first URL of the list is acquired which in most cases is the desired entity and its comment box is retrieved. If the entity linking through Wikipedia does not work, then it is considered that the entity was not found, and an empty list is returned.

The comment box of the input entity is compared with each comment box of the returned entities from ConceptNet, using the TF-IDF algorithm to extract semantic similarity, line 19. The rationale here is that the descriptions of two entities that are semantically related will contain common words. Therefore, the cosine similarity of the vectors that represent the two descriptions (after the TF-IDF algorithm) will be larger than the cosine similarity of two vectors that represent the description of two entities which are not related. TF-IDF was preferred in order not to raise the complexity of the framework using pretrained embedding vectors like Glove (Pennington *et al*., [2014](#)), Word2Vec (Rong, [2014](#)), or FastText (Joulin *et al*., [2016](#)). In this case, TF-IDF was used with stemming over the texts; stemming was applied when the texts were preprocessed, and thus words, which are not exactly the same, can be related.

TF-IDF can reduce the time complexity of Algorithm 1, as the complexity of TF-IDF is $O(n * log(n))$, where n is the number of words in both texts. On the other hand, word embeddings in Algorithm 1 could rise the time complexity at levels of class $O(n * m)$, where n is the number of words in the text, and m the number of vectors in the dataset of word embeddings. Looking at the complexities of TF-IDF and word embeddings one can see that TF-IDF is much quicker. More specifically, for the system to produce an answer using word embeddings, more than 3 minutes were needed approximately for its computation (Vassiliades *et al*., [2020](#), [2022](#)), which could not support an online question-answering system.

In order to define the semantic similarity between the entities, a new metric that is based on the combination of WordNet paths, TF-IDF scores, and ConceptNet weights was devised (Equation ([1](#))).

Algorithm 1 Semantic Matching Algorithm.

Require: entity
Ensure: hash_semantic_similarity

```
 1:  if entity in WordNet then
 2:      hash_property_values=get.ConceptNet_property_values(entity)
 3:      Comment_Box_Input = get.DBpediaCommentBox(entity)
 4:      hash_semantic_similarity = {}
 5:      for property in hash_property_values do
 6:          for value in hash_property_values [property] do
 7:              WordNet_Path = get.WordNet_Path(entity, value)
 8:              Comment_Box = Ø
 9:              if value in DBpedia then
10:                  Comment_Box = get.DBpediaCommentBox(value)
11:              end if
12:              if Comment_Box = Ø then
13:                  wiki_entity = get.WikipediaEntityURL(value)
14:                  Comment_Box = get.DBpediaCommentBox(wiki_entity)
15:              end if
16:              if Comment_Box = Ø then
17:                  continue
18:              end if
19:              TFIDF = TF-IDF(Comment_Box_Input, Comment_Box)
20:              hash_semantic_similarity[property] = (value, Sim(entity, value))
21:          end for
22:      end for

23:      hash_semantic_similarity = sorted(hash_semantic_similarity)
24: end if
```

This metric takes into consideration the smallest WordNet path, the ConceptNet weights, and the TF-IDF scores. TF-IDF and ConceptNet scores have a positive contribution to the semantic similarity of two words. On the other hand, the bigger the path is between two words in WordNet the smaller the semantic similarity is.

$$Sim\left(i, v_j\right) = \frac{\frac{1}{WNP\left(i, v_j\right)} + TFIDF\left(i, v_j\right) + CNW\left(i, p, v_j\right)}{3} \tag{1}$$

In (1), i is the entity given as input by the user, and v_j, for $j \in \mathcal{N}^*$, is each one of the different values returned from ConceptNet properties. $CNW\left(i, p, v_j\right)$ is the weight that ConceptNet gives for the triplet $\left(i, p, v_j\right)$, p stands for the property that connects i and v_j, and $0 \leq CNW(\cdot, \cdot, \cdot) \leq 1$. $TFIDF\left(i, v_j\right)$ is the score returned by the TF-IDF algorithm when comparing the DBpedia comment boxes of i and v_j, and $0 \leq TFIDF(\cdot, \cdot) \leq 1$. $WNP\left(i, v_j\right)$ is a 2-parameter function that returns the least common path between i and v_j, in the WordNet directed acyclic graph.

In case i and v_j have at least one common hypernym (ch), then the smallest path is acquired for the two words, whereas in case i and v_j, do not have a common hypernym (nch), their depths are added. Let, $depth\left(\cdot\right)$ be the function that returns the number of steps needed to reach from the root of WordNet to a given entity, then:

$$WNP\left(i, v_j\right) = \begin{cases} \min_{c \in C}\{depth(i) + depth\left(v_j\right) - 2 * depth(c)\} & ch \\ depth(i) + depth\left(v_j\right) & nch \end{cases}$$

where C is the set of common hypernyms for i and v_j, c is a common hypernym of i and v_j, and $min_{c \in C}\{\ldots\}$ returns the minimal value of the equation $depth(i) + depth(v_j) - 2 * depth(c)$ for $c \in C$. Also, $0 < WNP(\cdot, \cdot) \leq 1$.

Equation 1 can take scores in the range of (0,1], and the weights for $CNW(\cdot, \cdot, \cdot)$, $TFIDF(\cdot, \cdot)$, and $WNP(\cdot, \cdot)$ are equal to 1. Other variations of weights are possible, but they do not capture the synergy between the various metrics of the semantic relatedness of many pairs of entities that were examined. Moreover, the information from ConceptNet, WordNet, and DBpedia is used without any bias among them. Table 1 shows the values that Equation 1 returns for the pairs of Example 1. Notice that the values that the SMA extracts can either be incoming or out-coming to the input node. For instance, in Example 1 the *AtLocation* values are incoming to the park node, so the second line could be rather read tree, slides, and frisbees AtLocation park. The results are rounded to three decimals. The last step of the algorithm sorts the semantic similarity results of the returned entities with respect to the ConceptNet property and stores the new information into a hash map, line 23.

Example 1.

> *park **IsA:** city park, amusement park, national pary,*
> *park **AtLocation:** tree, slides, frisbees*
> *park **RelatedTo:** squirrel, bench, grass*

Pair	Relation	CNW	WNP	Sim
park-city park	*IsA*	*0.439*	*0.24*	*0.3395*
park-amusement park	*IsA*	*0.342*	*0.21*	*0.275*
park-national park	*IsA*	*0.213*	*0.26*	*0.2365*
park-tree	*AtLocation*	*0.23*	*0.378*	*0.3*
park-slides	*AtLocation*	*0.176*	*0.25*	*0.213*
park-frisbees	*AtLocation*	*0.265*	*0.128*	*0.1965*
park-squirrel	*RelatedTo*	*0.187*	*0.327*	*0.257*
park-bench	*RelatedTo*	*0.047*	*0.237*	*0.142*
park-grass	*RelatedTo*	*0.032*	*0.199*	*0.1155*

The evaluation of the Algorithm 1 was performed through a user evaluation since the argumentation dialogues presented in Section 4 were also using the SMA. The users were asked to evaluate the answers that the framework returned when using SMA, therefore, indicating if the quality of the semantic matching that the algorithm achieves is satisfactory or not.

4. The learning-through-argumentation mechanism of the framework

Learning-through-argumentation is the second component of the framework and is built upon the information retrieval component. The framework allows the user, after (s)he has received an answer to her/his question, to argue against the validity of the answers, if (s)he considers that there is an entity which is missing in the answers returned, or something is wrong and should not be part of the answers returned. The user must back up her/his opinion by indicating a trustworthy KB in which (s)he has found the information that (s)he supports. This component uses external knowledge from ConceptNet, and WordNet. If the user wins the argumentation dialogue, the argumentation component of the framework accepts that the information retrieval component has missed some entities, or it has returned something wrong in its answers, and it refines its KB to add the new information or delete existing knowledge. The workflow of the argumentation and learning component can be seen in Figure 4. Notice that Figure 4 is basically the extension of Figure 1, and the answers the user receives at the first step of Figure 4 are the answers
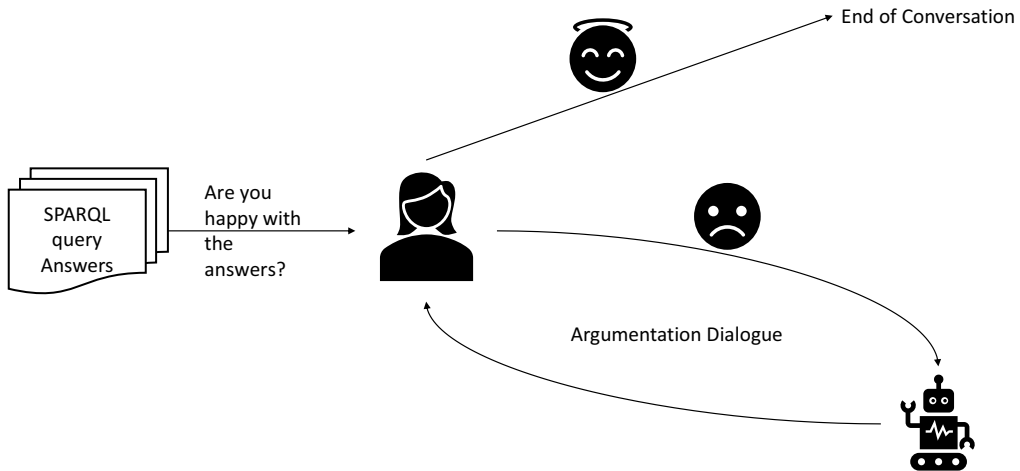
Figure 4. *Pipeline of the learning-through-argumentation mechanism of the framework*

returned by the framework in the last step of Figure 1. Also notice, that many details of Figure 1 are not shown in Figure 4, to focus mainly on the argumentation component. Each step in the workflow is annotated with a number in a circle that indicates the order in the argumentation dialogue. Finally notice, that there are alternative paths in the dialogue.

The main purpose of the learning-through-argumentation mechanism is to allow the framework to learn the preferences of each user, based on an argumentation dialogue through which the user will indicate specific preferences, and help the framework construct more user specific answers. The second purpose of the argumentation learning-through-argumentation mechanism is to gain the trust of the users though the explanations it provides in why it returned some results, and drive the users to gain more trust over the framework.

The pipeline for the learning-through-argumentation mechanism of the KG is straightforward, as the user as soon as (s)he receives the answer from the information retrieval mechanism it can argue over the systems answers. The interesting part is that according to which question the user initially asked the system, a new argumentation protocol will be triggered, in order for the user to argue with the system. If the argumentation procedure concludes in favor of the human, i.e., the system cannot defend it self anymore, the system will accept the users indication and will refine its internal KB either by deleting knowledge or adding new one.

Out of the 83 CQ which were gathered by domain experts (artist, designers and architects) in order to help construct the schema of the KG, 31 were formulated so that an argumentation dialogue could be performed on them. The reason for that is because some of the 83 could be clustered in a more generic query, for example the question *'Does the location L provokes stress if it contains the object X,Y,Z?'* could be also understood as *'Does the location L provokes stress?'* if considered in a more general manner. Below a list is given with all the question that were formulated so that the user could argue with the system over their results.

1. Which images might make me feel excited?
2. Which images might make me feel neutral?
3. Which images might make me feel sad?
4. Which images might make me feel happy?
5. The images which contain the X object what other object do they contain?
6. Are objects X,Y,Z related to landmarks?
7. The images which contain the X object as segmented object what feelings do they provoke?
8. The images which contain the X object as colourfulness object what feelings do they provoke?

9. Is the verge label X related with the object X,Y,Z?
10. With what feelings is the verge label X associated?
11. What can I add to the image with verge label X in order to make it more attractive?
12. What can I add to the image with verge label X in order to make it more usefull for the local citizens?
13. How are the locations associated to the stress of the human individuals?
14. Which one of the human individuals gets stressed more across the locations?
15. How do the object X,Y,Z affect the emotions of a user according to the location?
16. If I want the object X,Y,Z in an image what other object can I add in order to make it provoke happiness?
17. If I want the object X,Y,Z in an image what other object can I add in order to make it provoke sadness?
18. If I want the object X,Y,Z in an image what other object can I add in order to make it provoke excitment?
19. If I want the object X,Y,Z in an image what other object can I add in order to make it provoke boredom?
20. Is the phrase "..." related with the object X,Y,Z?
21. Is the phrase "..." related with the verge T?
22. Is the phrase "..." related with the sentiment happiness?
23. Is the phrase "..." related with the sentiment sadness?
24. Is the phrase "..." related with the sentiment excitment?
25. Is the phrase "..." related with the sentiment boredom?
26. Given the bounding box [...], what images in the location contain landmarks?
27. Given the bounding box [...], what images in the location contain the objects X,Y?
28. Given the bounding box [...], what images in the location make me happy?
29. Given the bounding box [...], what images in the location make me sad?
30. Given the bounding box [...], what images in the location make me excited?
31. Given the bounding box [...], what images in the location make me bored?

An argumentation protocol is showcased below to demonstrate how the framework learns new knowledge through the argumentation dialogue. In more detail all the steps of the argumentation dialogue that correspond to the Q11, are showcased. Notice that for Q11 the framework needs to propose objects that have been found in locations similar to X and in X, and provoke either happiness and/or arousal. The argumentation dialogue for the other queries work accordingly, with minor changes in the explanations that the system provides to the user.

- **Step 1:** The user addresses the query "What can I add to the image with verge label X in order to make it more attractive?"
- **Step 2:** The framework will return a list of objects that are found in places tagged with the label X and similar places and provoke happiness and/or arousal. Subsequently, the framework will ask back the user if (s)he agrees with its answer. If the user agrees, then the argumentation dialogue will end here in favor of the system, if the user disagrees the argumentation dialogues will proceed to **Step 3**. Notice that the answer may vary according to the question.
- **Step 3:** The system will then provide an explanation to the user on why it proposed these specific objects, and it would ask again the user if this explanation is adequate. If the user accepts the explanation, then the argumentation dialogue will end here in favor of the system, otherwise the argumentation dialogue will proceed to **Step 4**. Notice that the explanation may vary according to the question.
- **Step 4:** If the user is still not convinced, then the system will ask the user back if (s)he desires to change the thresholds that it takes into consideration to answer this question, in this case the segmentation and colourfulness of the objects in the images. If the user does not want to change

the threshold, then the system will accept that the user has won the argumentation dialogue and not relate again the objects with the verge label X. Otherwise, the system will ask again the user how much does (s)he want to reduce the restriction of the threshold, in this case by giving her/him 3 options Low, Medium, and High, and indicating that currently it was using the High threshold because it wanted to return the objects that provoke the most happiness and/or arousal. The High threshold is set to 75% for each object, the option Medium decreases the threshold to 50% and the option Low to 25%. In the case the user has chosen a threshold, then the system progresses to the final step **Step 5**.

- **Step 5:** If the user likes the new answers then the framework has won the argumentation dialogue, otherwise the user wins the argumentation dialogue, and the framework will not relate again the objects with the verge label X for this user.

Two notices need to be done here. First, the thresholds mentioned in Step 4, refer to the colourfulness and segmentation of each object in a image, these values are given by the visual analysis mechanism (see Subsection 3.1). So basically, when the user chooses a specific threshold then the learning-through-argumentation mechanism of the framework will consider the threshold indicated by the user when its going to add an entity in its answer. But notice that some questions might not have the option of changing a threshold.

The second notice is how the mechanism learns a preference. It is done through a log file for each user, for instance in the example below the mechanism will keep in a log file that the specific user, when choosing Q11 and input park, has the preference on the Low threshold value, so if the same user asks again the exact same question it will use the Low threshold instead of the default High threshold.

A more concrete example for this argumentation dialogue is shown below. In the following argumentation dialogue the suffix U is given to parts of the dialogue that correspond to the user and the suffix U to parts of the dialogue that correspond to the system.

- **U:** "What can I add to the image with verge label *park* in order to make it more attractive?"
- **S:** Well. . . I think you could add benches, fountains, statues, and squirrels (which are not objects). Do you like these options?
- **U:** No, I think I do not like them
- **S:** Ok, I related the objects benches, fountains, statues, and squirrels (which are not objects), with the verge label park, because in other figures tagged as park, amusement park, and national park, these were set to provoke the most happiness and/or arousal. Is this explanation adequate?
- **U:** No, it did not help me.
- **S:** Ok, do you want me to have a more flexible way of thinking over this question in order to see if these objects are indeed not related with the verge label that you gave me? If you want me to be more flexible then indicate me how much: Low, Medium, High. Currently I was using the High threshold.
- **U:** Low
- **S:** Ok great, apart from the aforementioned objects benches, fountains, statues, and squirrels, I could also recommend adding food kiosks, bike lanes, horses, ducks, sculptures, trees.

5. Evaluation

The evaluation of the KG was a three part procedure. In the first part, the consistency and completeness of the KG was evaluated; this was done with two different evaluation methods. First, the consistency of the KG was evaluated, by testing if it obeys a set of SHACL constrains (Subsection 5.1). Second, the completeness of the KG was performed by comparing the answers which the framework returned over the questions presented in Section 4, with the answers that a set of individuals gave for the same questions. In the second part, the evaluation of the information retrieval mechanism was performed by

computing the accuracy over the questions it could answer (Subsection 5.2). Finally, the third learning-through-argumentation mechanism was evaluated with a user-driven evaluation (Subsection 5.3).

5.1 Consistency and completeness of the knowledge graph

The consistency of the KG was evaluated through a validation procedure which inspected the syntactical and structural quality of the data which was mapped in the KG. More specifically, a set of SHACL shape rules was developed to check if the data which was mapped respected the data types of the properties, as well as the cardinality of each property. One SHACL rule was developed for each class in the KG, ending up to set of 21 SHACL rules.

The purpose of each rule, was to check for the properties attached to the class if there are any violations regarding: (i) The data types of the properties, for example if the property was expecting string values and an integer was mapped as value of the property, (ii) the cardinality of the properties, for example a property has a max 1 cardinality and 2 values were mapped for the same instance, and (iii) for the object type properties if the object values of the property belong to the range class of the property. One can see an exemplary SHACL rule below for the class *:SemSegContainer*. Fortunately, no violations were returned to any rule.

```
          SHACL Rule for Class :SemSegContainer
:SemSegContainerShape
  a sh:NodeShape ;
  sh:targetClass :SemSegContainer ;
sh:property [
    sh:path :isSemSegLabel ;
    sh:maxCount 1 ;
    sh:name "is sem seg label" ;
    sh:class :SemSegLabel ;
] ;
sh:property [
    sh:path :hasAccuracy ;
    sh:name "has accuracy" ;
    sh:datatype xsd:double ;
] ;
sh:property [
    sh:path :hasPercentage ;
    sh:name "has percentage" ;
    sh:datatype xsd:double ;
] ;
sh:property [
    sh:path :semseg_ta_contr_imageability ;
    sh:name "has sem seg ta contr imageability" ;
    sh:datatype xsd:float ;
] .
```

Next, the completeness of the KG was evaluated through a user evaluation. We asked from a group of 20 individuals to compute their own queries and give 10 answers for each question, following the schema of questions presented in Section 4. Notice that these questions are basically the CQs presented in subsection 3.3. Therefore, each individual had to compute 31 questions and give 10 answers for each one of them, ending up in a gold-standard dataset of 200 answers per question.

Table 1. *Accuracy of the framework for each question*

Question	Accuracy	Accuracy percentage
Q1	166/200	83%
Q2	153/200	76,5%
Q3	158/200	79%
Q4	161/200	80,5%
Q5	165/200	82.5%
Q6	161/200	80,5%
Q7	158/200	79%
Q8	153/200	76,5%
Q9	150/200	75%
Q10	154/200	77%
Q11	154/200	77%
Q12	158/200	79%
Q13	166/200	83%
Q14	153/200	76,5%
Q15	158/200	79%
Q16	161/200	80,5%
Q17	166/200	83%
Q18	161/200	80,5%
Q19	158/200	79%
Q20	153/200	76,5%
Q21	150/200	75%
Q22	154/200	77%
Q23	154/200	77%
Q24	154/200	77%
Q25	159/200	79,5%
Q26	157/200	78,5%
Q27	157/200	78,5%
Q28	160/200	80%
Q29	158/200	79%
Q30	153/200	76,5%
Q31	150/200	75%
Average	4873/6200	78,5%

Subsequently, we compared the answers of the users with the answers that our framework returned for the same questions, and if there were common answers these would be considered as accurately predicted. For example, if the framework has 5 common answers with a user for a specific query, then we would consider that the framework would have accurately predicted 5 answers. One can see the accurately predicted answers per questions in Table 1. Moreover, notice that some of the individuals in this evaluation were the same with the individuals of the evaluation of the learning-through-argumentation mechanism, in subsection 5.3.

5.2 Information retrieval mechanism evaluation

The evaluation of the information retrieval mechanism was conducted using the accuracy of the information retrieval mechanism (Equation 2), over the two real-life scenarios presented in Section 3.3 (i.e.,

Table 2. *Accuracy for Scenario A and Scenario B*

	Accuracy
Scenario A	0.9649
Scenario B	0.9632

Scenario A and Scenario B).

$$accuracy = \frac{|\{Number\_of\_Answered\_Queries\}|}{|\{Number\_of\_Total\_Queries\}|} \tag{2}$$

The dataset on which the information retrieval mechanism was evaluated contains a set of 1200 images for each scenario and can be found here[1]. For Scenario A, the number of answered queries for images were 1157, while for Scenario B were 1142.

Based on the aforementioned numbers the accuracy for both scenarios can be found in Table 2. Notice, the results are rounded to four decimals.

5.3 Learning-through-argumentation mechanism evaluation

The argumentation mechanism is an automated add-on pipeline whose purpose is to assess the user's opinions on the system architecture's results. By engaging the users into discussions with a generative language model, on the one hand, the system's results are evaluated directly by a human agent, and on the other hand descriptions are provided to the user to help him/her understand the procedures adopted to produce such results. The evaluation of the learning-through-argumentation mechanism was to let the users argue with the framework over the results returned by the questions presented in Section 4.

The pipeline for the evaluation of the learning-through-argumentation was the users to handpick question from the set of question we present in Section 4, and argue over the returned results. We asked each user to do at least 15 argumentation dialogues with the framework, which is what happened, but no one would go to more than 15 argumentation dialogues. In total 31 individuals took part in these evaluations and the demographics of these individuals are the following. The participants were 78% males and 22% females and 81% were less than 34 years old. 66% had completed a master's degree and 41% were either married or cohabiting with a significant other. 31% worked within the creative industries while 13% was unsure. 88% expressed confidence on their ability to read, comprehend, and write in English while 87% expressed confidence in learning and using new technologies. Only 22% expressed confidence in distinguishing a text generated by a human expert from a chatbot while on the contrary 34% did not feel confident.

From the beginning until the closure, 7 distinct dialogues were read by questionnaire participants. After each argumentation dialogue, the participants were asked how satisfied they felt about the quality of the argumentation in a Likert scale from 1 to 5, ranging from very satisfied to very dissatisfied. The average percentages results of the first question type are illustrated in Figure 5. Some general assumptions imply that almost 70% of the participatory body felt at least satisfied about the quality of the argumentation dialogue.

6. Discussion and conclusion

In this paper, a KG for describing emotions related to and practical design for, the indoor and urban adaptive design was presented. The KG is packed with emotional, physiological, visual, and linguistic measurements, allowing the building of adaptable settings. It is targeted at artists, designers, and architects, and its goals include redesigning indoor work spaces to make employees feel more comfortable and enhancing the functionality, mobility, and aesthetic appeal of spaces while respecting the environment
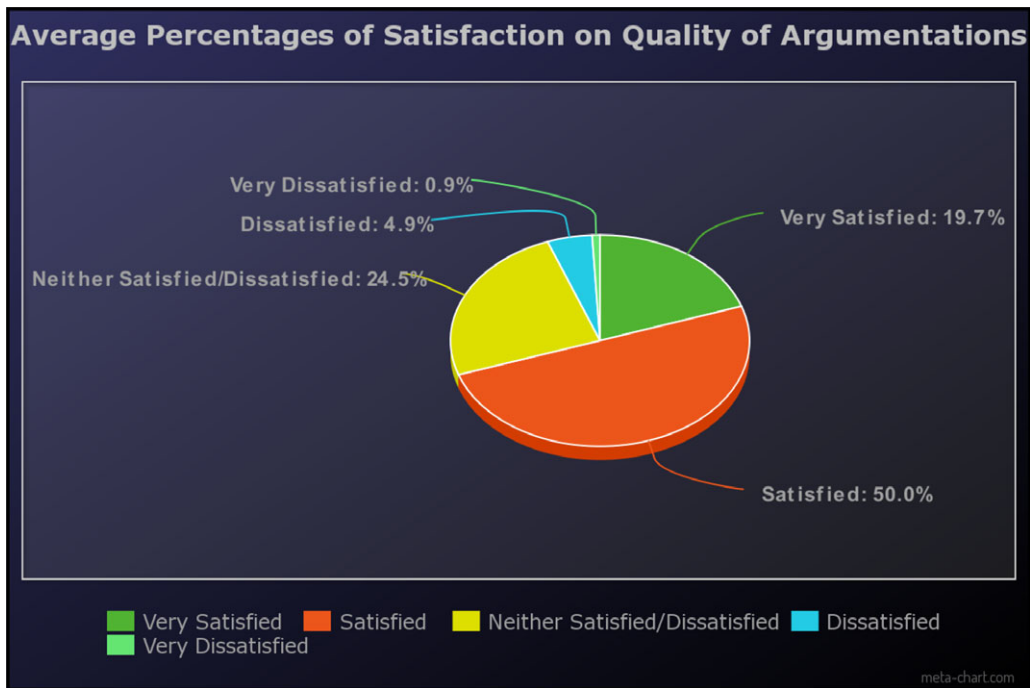
Figure 5. *How satisfied do you feel about the quality of the argumentation?*

and cultural heritage. However, other users can also utilize and take advantage of this KG. For instance, ideas from city residents might be quite helpful when revamping urban areas. Additionally, the KG provides a straightforward approach to retrieve the data inside through an information retrieval mechanism. This is why it retrieves information that is useful in real-world situations. Experts contributed the scenarios that were used to create the information retrieval mechanism.

The user can access all the knowledge inside the KG through a set of predefined SPARQL query templates. The user can also address ad hoc SPARQL queries to the KG. Additionally, by creating a semantic-matching algorithm that uses semantic information from WordNet to detect semantic similarity between entities existing in the framework's internal KB and entities from the KG of ConceptNet, the scope of queries the framework may respond to was expanded.

The learning-through-argumentation mechanism's primary goal is to enable the framework to understand each user's preferences by allowing it to engage in an argumentation conversation where the user can express particular preferences and assist the framework in creating additional replies that are tailored to each individual user. The intention was to create a mechanism that would refine the internal KB of the framework, with a method closer to the human approach of reasoning. As humans usually obtain new knowledge through argumentation, when their opinion is proven wrong or there is some missing knowledge. The learning-through-argumentation mechanism's second goal is to increase users' trust in the framework by gaining their confidence through the explanations it offers for why it delivered certain results. Therefore, it was intended to help the user understand why the system has answered a question in a specific manner, in other words the intention was to increase the explainability and transparency of the system on how it takes decisions.

For the creation of surroundings that can adapt, the KG is filled with emotional, physiological, visual, and textual metrics. It is primarily targeted at artists, designers, and architects, and its goals include, on one hand, redesigning indoor work spaces to make employees feel more comfortable and, on the other hand, enhancing the functionality, mobility, and esthetic appeal of spaces while respecting the environment and cultural heritage. But, other users can also make use of the KG. For instance, ideas

from city residents might be quite helpful when revamping urban areas. Additionally, the KG provides a straightforward approach to retrieve the data through an information retrieval mechanism. This is why it retrieves information that is useful in real-world situations. Experts contributed in the generation of the scenarios that were used to create the information retrieval mechanism.

The KG's consistency, completeness, and information retrieval mechanism were assessed as the last step. Using CQs that were gathered by domain experts, the KG's completeness was assessed, by comparing the answers of the framework with the answers given to the same questions by a group of individuals. The consistency of the KG was evaluated through a set of SHACL rules, from which none returned any violation to the constrains.

Concerning the information retrieval mechanism, both Scenario A (98.21%) and Scenario B (98.12%) received high accuracy scores, demonstrating that each scenario may be employed as a standalone mechanism for assisting users by offering useful information. Furthermore, it can be said that the non answered occurrences that were overlooked in Scenario A and Scenario B were due to the fact that both of these scenarios call for an image as an input, which is then examined and the Top-3 segmented labels (i.e., the Top-3 object labels that are most likely present in the image) are taken into account in order to locate related images from the KG. As a result, the images for which the framework could not answer had Top-3 segmented labels that were not presented simultaneously in any image in the KG.

Three outcomes stand out from the framework's knowledge retrieval component evaluation. The framework can be considered as a baseline for knowledge representation for environments that have to do with emotions related and urban adaptive design as a primary (or secondary) source of knowledge, because of the high percentage of subjects who are satisfied with the knowledge retrieval component's answers. The second technique of evaluation suggests that the information retrieval component, can be utilized as a benchmark for evaluating other cognitive robotic systems functioning in an environment, that is interested in the emotions pertinent to an urban or indoor location.

On the other hand, the evaluation of the framework's learning-through-argumentation component reveals that the users find the argumentation dialogue that the component generates to be logical and persuasive. The component largely succeeds in achieving its goals as all arguments scenarios received excellent satisfaction ratings.

The two components of this study's limitations are the following. First, only the context of emotionally relevant and urban adaptive design is now available for the knowledge retrieval component. The ontology should be expanded with new classes and relations to represent the new knowledge in order to broaden the framework's scope. Another drawback of the Semantic Similarity Algorithm is that, although being employed in a 'controlled way' the information it gives still has a lot of noise. As a result, additional advanced techniques (in addition to the suggested metric) should be created to filter the results that are returned. Furthermore, humans may argue in a wide variety of ways, but the learning-through-argumentation component currently only supports a limited number of cases. More scenarios—or rather, a flexible dialogue protocol that can accommodate various arguments scenarios—should be created in the future. The most well-known argumentation framework, Dung (1995), is now the foundation for the learning-through-argumentation component. Yet, because it is abstract, there is not much room for explanation. Other argumentation frameworks, like the Structured Argumentation Framework (Modgil & Prakken, 2014) or the Argumentation Framework with Domain Assignments (Vassiliades *et al*., 2021), could be taken into consideration to compute arguments in order to get around this limitation.

Regarding upcoming work, the intention is to assess and contrast the knowledge in the KG with other KGs already in existence which are relative to indoor and urban adaptable design. It is also interesting looking into and increase the range of real-world scenarios that the information retrieval mechanism can accommodate. Since the queries are now written in SPARQL, the intention is to provide a user interface that is friendlier. The semantic matching algorithm will also be expanded by including data from more ontologies. Finally, by utilizing techniques from the semantic question answering (Antoniou & Bassiliades, 2022) study field, the system might be expanded to accept queries in natural language.

These techniques enable the translation of questions from natural language to SPARQL query templates, which may then be directed towards the underlying knowledge network(s).

References

Akhdinirwanto, R., Agustini, R. & Jatmiko, B. 2020. Problem-based learning with argumentation as a hypothetical model to increase the critical thinking skills for junior high school students. *Jurnal Pendidikan IPA Indonesia* **9**(3), 340–350.

Aleven, V. & Ashley, K. D. 1997. Evaluating a learning environment for case-based argumentation skills. In *Proceedings of the 6th International Conference on Artificial Intelligence and Law*, 170–179.

Andreadis, S., Moumtzidou, A., Apostolidis, K., Gkountakos, K., Galanopoulos, D., Michail, E., Gialampoukidis, I., Vrochidis, S., Mezaris, V. & Kompatsiaris, I. 2020. Verge in vbs 2020. In *International Conference on Multimedia Modeling*. Springer, 778–783.

Antoniou, C. & Bassiliades, N. 2022. A survey on semantic question answering systems. *The Knowledge Engineering Review* **37**, e2.

Ashley, K. D., Desai, R. & Levine, J. M. 2002. Teaching case-based argumentation concepts using dialectic arguments vs. didactic explanations. In *Intelligent Tutoring Systems: 6th International Conference, ITS 2002 Biarritz, France and San Sebastian, Spain, June 2–7, 2002 Proceedings*. Springer, 585–595.

Asim, M. N., Wasim, M., Khan, M. U. G., Mahmood, N. & Mahmood, W. 2019. The use of ontology in retrieval: a study on textual, multilingual, and multimedia retrieval. *IEEE Access* **7**, 21662–21686.

Berland, L. K. & McNeill, K. L. 2010. A learning progression for scientific argumentation: Understanding student work and designing supportive instructional contexts. *Science Education* **94**(5), 765–793.

Cayrol, C., de Saint-Cyr, F. D. & Lagasquie-Schiex, M.-C. 2008. Revision of an argumentation system. *KR* **2008**, 124–134.

Chen, Q., Bragg, J., Chilton, L. B. & Weld, D. S. 2019. Cicero: Multi-turn, contextual argumentation for accurate crowdsourcing. In *Proceedings of the 2019 Chi Conference on Human Factors in Computing Systems*, 1–14.

Chernova, S., Chu, V., Daruna, A., Garrison, H., Hahn, M., Khante, P., Liu, W. & Thomaz, A. 2020. Situated bayesian reasoning framework for robots operating in diverse everyday environments. In *Robotics Research*. Springer, 353–369.

Chi, N.-W., Jin, Y.-H. & Hsieh, S.-H. 2019. Developing base domain ontology from a reference collection to aid information retrieval. *Automation in Construction* **100**, 180–189.

Clark, D. B., Sampson, V., Weinberger, A. & Erkens, G. 2007. Analytic frameworks for assessing dialogic argumentation in online learning environments. *Educational Psychology Review* **19**, 343–374.

Coste-Marquis, S., Konieczny, S., Mailly, J.-G. & Marquis, P. 2014a. On the revision of argumentation systems: Minimal change of arguments statuses. In *Fourteenth International Conference on the Principles of Knowledge Representation and Reasoning*.

Coste-Marquis, S., Konieczny, S., Mailly, J.-G. & Marquis, P. 2014b. A translation-based approach for revision of argumentation frameworks. In *Logics in Artificial Intelligence: 14th European Conference, JELIA 2014, Funchal, Madeira, Portugal, September 24–26, 2014. Proceedings 14*. Springer, 397–411.

Čyras, K., Satoh, K. & Toni, F. 2016. Explanation for case-based reasoning via abstract argumentation. In *Computational Models of Argument*. IOS Press, 243–254.

Drapeau, R., Chilton, L., Bragg, J. & Weld, D. 2016. Microtalk: Using argumentation to improve crowdsourcing accuracy. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 4, 32–41.

Dung, P. M. 1995. An argumentation-theoretic foundation for logic programming. *The Journal of Logic Programming* **22**(2), 151–177.

Falappa, M. A., Kern-Isberner, G. & Simari, G. R. 2009. Belief revision and argumentation theory. In Argumentation in Artificial Intelligence, 341–360.

Fan, X., Craven, R., Singer, R., Toni, F. & Williams, M. 2013. Assumption-based argumentation for decision-making with preferences: a medical case study. In *Computational Logic in Multi-Agent Systems: 14th International Workshop, CLIMA XIV, Corunna, Spain, September 16–18, 2013. Proceedings 14*. Springer, 374–390.

Fan, X. & Toni, F. 2015. On explanations for non-acceptable arguments. In *International Workshop on Theory and Applications of Formal Argumentation*. Springer, 112–127.

Fellbaum, C. 2010. Wordnet. In *Theory and Applications of Ontology: Computer Applications*. Springer, 231–243.

Fiorini, R. A. 2020. Computational intelligence from autonomous system to super-smart society and beyond. *International Journal of Software Science and Computational Intelligence (IJSSCI)* **12**(3), 1–13.

Heras, S., Jordán, J., Botti, V. & Julián, V. 2013. Argue to agree: a case-based argumentation approach. *International Journal of Approximate Reasoning* **54**(1), 82–108.

Hu, B., Gaurav, A., Choi, C. & Almomani, A. 2022. Evaluation and comparative analysis of semantic web-based strategies for enhancing educational system development. *International Journal on Semantic Web and Information Systems (IJSWIS)* **18**(1), 1–14.

Hyvönen, E. 2012. Publishing and using cultural heritage linked data on the semantic web. *Synthesis Lectures on the Semantic Web: Theory and Technology* **2**(1), 1–159.

Icarte, R. T., Baier, J. A., Ruz, C. & Soto, A. 2017. How a general-purpose commonsense ontology can improve performance of learning-based image retrieval. arXiv preprint arXiv:1705.08844.

Joulin, A., Grave, E., Bojanowski, P., Douze, M., Jégou, H. & Mikolov, T. 2016. Fasttext. zip: Compressing text classification models. arXiv preprint arXiv:1612.03651.

Kelly, K. T. 1998. The learning power of belief revision. In *TARK*, **98**. Citeseer, 111–124.

Kumar, V. R. S., Khamis, A., Fiorini, S., Carbonera, J. L., Alarcos, A. O., Habib, M., Goncalves, P., Li, H. & Olszewska, J. I. 2019. Ontologies for industry 4.0. *The Knowledge Engineering Review* **34**, e17, 1 of 14.

Lopes, D. M. 2007. Shikinen sengu and the ontology of architecture in japan. *The Journal of Aesthetics and Art Criticism* **65**(1), 77–84.

Lynch, K. et al. 1960. The image of the city (vol. 11).

Modgil, S. & Prakken, H. 2014. The aspic+ framework for structured argumentation: a tutorial. *Argument & Computation* **5**(1), 31–62.

Možina, M., Žabkar, J., Bench-Capon, T. & Bratko, I. 2005. Argument based machine learning applied to law. *Artificial Intelligence and Law* **13**, 53–73.

Možina, M., Žabkar, J. & Bratko, I. 2007. Argument based machine learning. *Artificial Intelligence* **171**(10–15), 922–937.

Munir, K. & Anjum, M. S. 2018. The use of ontologies for effective knowledge modelling and information retrieval. *Applied Computing and Informatics* **14**(2), 116–126.

Okuno, K. & Takahashi, K. 2009. Argumentation system with changes of an agent's knowledge base. In *Twenty-First International Joint Conference on Artificial Intelligence*. Citeseer.

Ontañón, S. & Plaza, E. 2007. Learning and joint deliberation through argumentation in multiagent systems. In *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems*, 1–8.

Ontañón, S. & Plaza, E. 2010. Multiagent inductive learning: an argumentation-based approach. In *ICML*, 839–846.

Pennington, J., Socher, R. & Manning, C. D. 2014. Glove: global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543.

Pilotti, P., Casali, A. & Chesnevar, C. 2012. A belief revision approach for argumentation-based negotiation with cooperative agents. In *9th International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2012), Valencia, Spain*. Citeseer.

Pilotti, P., Casali, A. & Chesnevar, C. 2014. Incorporating object features in collaborative argumentation-based negotiation agents. In *Brazilian Conference on Intelligent Systems (BRACIS)/Encontro Nacional de Inteligencia Artificial e Computacional (ENIAC)*, Sao Carlos, SP, Brazil, 31–37.

Pistola, T., Georgakopoulou, N., Shvets, A., Chatzistavros, K., Xefteris, V.-R., Garca, A. T., Koulalis, I., Diplaris, S., Wanner, L., Vrochidis, S. et al. 2022. Imageability-based multi-modal analysis of urban environments for architects and artists. In *International Conference on Image Analysis and Processing*. Springer, 198–209.

Qiu, W., Li, W., Liu, X. & Huang, X. 2021. Subjective street scene perceptions for shanghai with large-scale application of computer vision and machine learning, Technical report, EasyChair.

Rahwan, I., Moraitis, P. & Reed, C. 2005. *Argumentation in Multi-Agent Systems: First International Workshop, ArgMAS 2004, New York, NY, USA, July 19, 2004, Revised Selected and Invited Papers*, 3366. Springer.

Raven, D., de Boer, V., Esmeijer, E. & Oomen, J. 2020. Modeling ontologies for individual artists. Vrije Universiteit Amsterdam.

Rong, X. & 2014. word2vec parameter learning explained. arXiv preprint arXiv:1411.2738.

Schneider, A. 2020. Alternatives: World ontologies and dialogues between contemporary arts and anthropologies. In *Alternative Art and Anthropology*. Routledge, 1–26.

Slonim, N., Bilu, Y., Alzate, C., Bar-Haim, R., Bogin, B., Bonin, F., Choshen, L., Cohen-Karlik, E., Dankin, L., Edelstein, L. et al. 2021. An autonomous debating system. *Nature* **591**(7850), 379–384.

Speer, R., Chin, J. & Havasi, C. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-first AAAI Conference on Artificial Intelligence*.

Stathopoulos, E., Vassiliades, A., Diplaris, S., Vrochidis, S., Bassiliades, N. & Kompatsiaris, I. 2023. The mindspaces knowledge graph: Applied logic and semantics on indoor and urban adaptive design. In *Proceedings of the 15th International Conference on Agents and Artificial Intelligence - Volume 3: ICAART*. INSTICC, SciTePress, 334–341.

Vassiliades, A., Bassiliades, N., Gouidis, F. & Patkos, T. 2020. A knowledge retrieval framework for household objects and actions with external knowledge. In *International Conference on Semantic Systems*. Springer, 36–52.

Vassiliades, A., Bassiliades, N. & Patkos, T. 2021. Argumentation and explainable artificial intelligence: a survey. *The Knowledge Engineering Review* **36**, e5.

Vassiliades, A., Bassiliades, N., Patkos, T. & Vrakas, D. 2022. An open-ended web knowledge retrieval framework for the household domain with explanation and learning through argumentation. *International Journal on Semantic Web and Information Systems (IJSWIS)* **18**(1), 1–34.

Vassiliades, A., Patkos, T., Flouris, G., Bikakis, A., Bassiliades, N. & Plexousakis, D. 2021. Abstract argumentation frameworks with domain assignments. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21)*. IJCAI: International Joint Conferences on Artificial Intelligence Organization, 2076–2082.

Veerman, A. L. 2000. *Computer-Supported Collaborative Learning through Argumentation*. PhD thesis, Urtecht University.

Von Aufschnaiter, C., Erduran, S., Osborne, J. & Simon, S. 2008. Arguing to learn and learning to argue: Case studies of how students' argumentation relates to their scientific knowledge. *Journal of Research in Science Teaching: The Official Journal of the National Association for Research in Science Teaching* **45**(1), 101–131.

Wagner, A. & Rüppel, U. 2019. Bpo: The building product ontology for assembled products. In *Proceedings of the 7th Linked Data in Architecture and Construction Workshop (LDAC 2019)*, Lisbon, Portugal, 12.

Young, J., Basile, V., Kunze, L., Cabrio, E. & Hawes, N. 2016. Towards lifelong object learning by integrating situated robot perception and semantic web mining. In *Proceedings of the Twenty-second European Conference on Artificial Intelligence*. IOS Press, 1458–1466.

Young, J., Basile, V., Suchi, M., Kunze, L., Hawes, N., Vincze, M. & Caputo, B. 2017. Making sense of indoor spaces using semantic web mining and situated robot perception. In *European Semantic Web Conference*. Springer, 299–313.

Yu, B. 2019. Research on information retrieval model based on ontology. *EURASIP Journal on Wireless Communications and Networking* **2019**(1), 1–8.

Zamazal, O. 2020. A survey of ontology benchmarks for semantic web ontology tools. *International Journal on Semantic Web and Information Systems (IJSWIS)* **16**(1), 47–68.