Architecting Dietary Intake Monitoring as a Service Combining NLP and IoT

Preprint

Antonio Benítez-Guijarro, Zoraida Callejas, Manuel Noguera, Kawtar Benghazi
Software Engineering Department, University of Granada, Spain
{ajbenitez, zoraida, mnoguera, benghazi}@ugr.es

ABSTRACT

Currently there exist many tools that support monitoring and encouragement of healthy nutrition habits in the context of wellness promotion. In this domain, interfaces based on natural language provide more flexibility for nutritional self-reporting than traditional form-based applications, allowing the users to provide richer and spontaneous descriptions. Nonetheless, in certain circumstances, natural language records may miss some important aspects, such as the quantity of food eaten, which results in incomplete recordings. In the Internet-of-Things (IoT) paradigm, smart home appliances can support and complement the recording process so as to make it more accurate. However, in order to build systems that support the semantic analysis of nutritional self-reports, it is necessary to integrate multiple inter-related components, possibly within complex e-health platforms. For this reason, these components should be designed and encapsulated avoiding monolithic approaches that derive in rigidity and dependency of particular technologies. Currently, there are no models or architectures that serve as a reference for developers towards this objective. In this paper, we present a service-based architecture that helps to contrast and complement the descriptions of food intakes by means of connected smart home devices, coordinating all the stages during the process of recognizing food records provided in natural language. Additionally, we aim to identify and design the essential services that are required to automate the recording and subsequent processing of natural language descriptions of nutritional intakes in association with smart home devices. The functionalities provided by each of these services are ready to work in isolation, just out of the box, or in downstream pipeline processes, bypassing the inconveniences of monolithic architectures.

1 Introduction

Nowadays, there is an increase in the development and use of applications designed to control and encourage healthy habits [1, 2]. These applications generally rely on cloud-based telemonitoring systems that store and process the data about users who are being monitored [3]. Telemonitoring systems allow users to analyze their habits, recognizing unhealthy behaviors and then planning and suggesting actions to improve their well-being.

A balanced nutrition, providing all essential nutrients in the right amounts, is the cornerstone of good health and disease prevention. There are many studies and applications that offer methods that help users to manage their diet on the basis of self-reports of food intakes [4]. With these methods, users usually select their meals from a predefined list or type of dish recipes, and the system calculates the nutritional information of each record.

However, these interfaces may be cumbersome to work with for some users and may lead to low adherence [5]. Using a voice or textual interface, users can dictate the list of foods consumed and interact more intuitively [6]. When the user interacts and provides a natural language description of the ingested meal, this description can be processed and decomposed identifying the essential parameters to compute a nutritional record [7].

Although this approach is sound and works fairly well [8], it suffers from the uncertainty that is inherent to self-reporting approaches: the reports are usually inaccurate or incomplete due to different factors, including low adherence, subjective biases and users' competence [9, 10]. Information available from smart devices in IoT settings could be leveraged to complement and improve the records about users' intakes. At present, several major companies such as Philips, Amazon and some others, are developing smart devices for home task automation, self-monitoring and habits awareness, in order to make *smartness* more pervasive in daily environments. These devices bring new ways to collect information about users and their daily eating habits. For example, many of these new devices, such as cooking weight scales, offer the possibility to collect and send weight data [11]; while others, such as smart fridges, can be aware of their current contents [12]. There are also smart dishes which can provide additional information about the size (volume and weight) of the rations they contain. In this work, we will refer indistinctly to this kind of connected devices, which sensed information can be accessed through the Internet (directly or through a proxy), as smart home appliances and smart kitchenware devices [13]. All these novel technologies bring about underexplored possibilities to record, analyze, monitor and supervise users nutrition.

Figure 1 presents a general overview of the different technologies, steps and elements usually involved in nutrition monitoring systems. As it can be observed, smart settings for food intake monitoring through natural language interfaces must provide support to plenty of different services, such as request handling, data transformation and services for computing nutritional information, normally in a downstream pipeline sequence, as well as specific information models and structures. If natural language processing (NLP) is usually a complex software task, the extraction of nutritional information from users' utterances adds extra complexity to the development of these systems. At present, the providers of popular interfaces based on natural language cannot meet the increasing demands from software developers, so modular approaches and solutions are rather advisable, allowing specific algorithms for each processing stage to be evolved and improved independently. Actually, the identification of nutritional information, by itself, is carried out through different processing steps and makes use of different approaches with dissimilar efficacy, being those based on keywords, statistical analysis and structured analysis, the most common ones. However, most of the existing proposals present a monolithic, rigid structure, so that the developer is forced to consume all the services or none [14]. That is, developers cannot benefit from only part of the natural language processing from one provider and search for other services that could fit their demands better. Moreover, there do not exist explicit proposals that can serve of guidance and reference to implement this type of service platforms and systems.

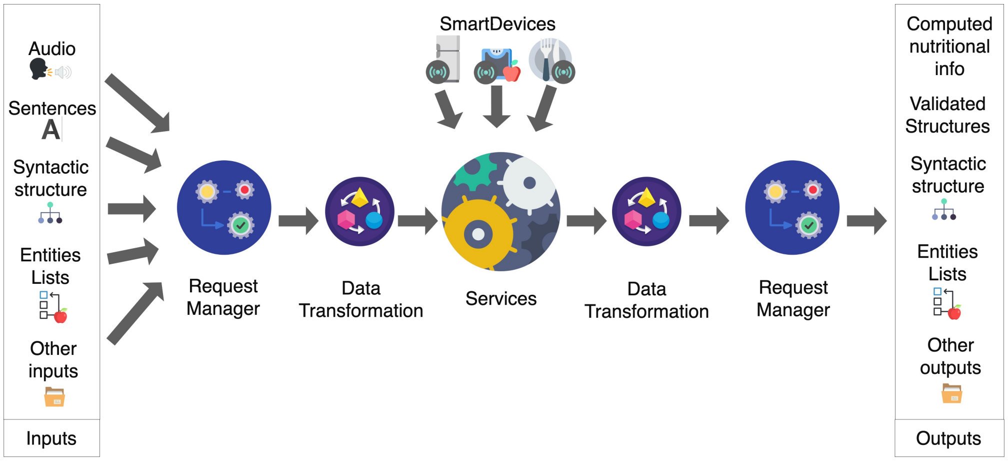


Figure 1: General overview of inputs, components and services for nutrition monitoring through oral interfaces and smart devices.

In this paper, we introduce a service-oriented architecture for dietary intake monitoring encompassing a set of independent services ready to work in isolation, right out of the box, or in downstream pipeline processes. These services provide the necessary information to carry out nutritional monitoring based on natural language inputs that are interpreted using semantic parsing and complemented with information retrieved from IoTs smart devices.

The architecture aims to: 1) support dietary intake monitoring in natural language, 2) contrast and complement the nutritional descriptions of the records in environments where users will use connected devices for cooking and eating; and 3) provide guidance and support to the development of new applications for nutritional monitoring based on services.

The rest of the paper is organized as follows. Section 2 presents the current approaches and related work for architectures to support connected devices and natural language processing in the nutrition domain. Section 3 describes the design principles that have guided the proposed architecture, which is presented in Section 4 and illustrated with an example in Section 5. Finally Section 6 presents the conclusions and future work.

2 Related work

According to recent reports, there are more than 300k mHealth apps in the market [15] and most of them are related to fitness and nutrition. Although many do not have validity for becoming “prescriptible” by health professionals [15], there exists scientific evidence that their proliferation is beneficial and can support users to manage their well-being if they are implemented appropriately, with enough accuracy and efficiency [16]. In addition, in order to reveal the users’ dietary intake, it is required that the apps engage them with regular self-monitoring. This can only be possible if the interface of the applications are easy to use.

The time required for record tracking, the type and ease of using the device, and other factors have an impact in adherence to dietary self-monitoring [17], and natural language interfaces reduce recording times and foster engagement [18].

When using traditional nutrition tracking methods such meal lists and photographs, users must select a predefined meal and size which may not match what they actually ate. Previous studies with multiple nutrition apps have described how users encounter numerous challenges in the recording of mixed dishes (e.g., lasagna) and home-made foods, which they describe as a “tedious, time-consuming, and difficult” process [19]. With natural language interfaces, the users can describe the meals with detail and indicate the ingredients and portions with their own words, but it is necessary to overcome its inherent ambiguity.

Smart environments have an increased awareness of the users' habits through IoT by integrating new interconnected home appliances [20, 21, 22]. All these devices transparently obtain pertinent information that can complement the knowledge that is extracted from natural language through direct explicit interaction with the user [23].

As highlighted in [24], there is a lack of architectures to promote healthy lifestyles that are not purely market-oriented. In particular, the combination of IoT devices with natural language interaction is even less common and there are no standard reference models and architectures that incorporate all these technologies.

As argued in [25], the system architecture is an important enabler for many properties of a product, from which they highlight composability, deployability, stability, and configurability. Apart from these general properties, the authors emphasize that in certain domains, consistent user interfaces are a key aspect, as it is our case. Initial work on models and architectures for smart environments describe their general requirements and ideal properties and how information from sensors and devices can be considered to provide context-awareness [26, 27, 28]. Despite these models specify some of the general requirements, these works do not address how context-awareness can be helpful tools for nutrition monitoring.

The mutual influence of interface and software architecture engineering aspects has not been addressed in depth in the literature. In [29], the authors present a study of the current state of the art in this topic and they conclude that there are not many software architecture proposals that deal with context-awareness in an explicit way and that there exist human-computer interaction (HCI) pieces of work that focus on context-aware adaptations, but they neglect partially or completely the adaptation processes that could be accomplished using the information retrieved (e.g. from smart devices). The authors emphasize this is a significant gap that should be covered in future research addressing adaptation at architectural (configuration of architectural components) and HCI (with a flexible user interface) levels.

Regarding the architectural configuration to automatically convert patient dietary records into nutrient components, nutrition assessment can be streamlined using different services - or even apps as suggested in [16]. However, these have not been identified in the literature, which usually provides ad-hoc solutions or present general "ecosystems" [30].

Service-oriented software architectures for IoT are commonplace. For example, in [31] the relevance of the service to be used at each moment includes information retrieved from the user's context through IoT devices and sensors. Other architectures for IoT are "semantic driven", although semantics is used to represent the data exchanged and not as a service to be used for processing natural language inputs. This is the case of the architecture presented in [32] for the healthcare domain.

On the other hand, the architectures that process natural language in the nutritional context, either as an explicit user input or in the form of texts (e.g. recipes and online recommendations), are focused on the different components involved for training, testing and performing the natural language processing [33], but not on how they can be integrated into an architecture that is extensible.

Table 1 summarizes the contributions of related work in relation to the main aspects addressed in this paper and that we have identified as to provide an *Explicit Architecture* description, support for *Spoken Interaction* and *Nutritional Data Processing*, *Complement of Records with Info from IoT Devices*, architecture *Modularity* and *Extendability*, as well as the degree of *Service Coupling*.

Table 1: Comparison of related work contributions.

Proposals	Explicit Architecture	Spoken Interaction	Nutritional Data Processing	Records Complement with Info from IoT Devices	Modularity	Extendability	Service Coupling
[20]	Partially	Partially	X	Partially	X	X	X
[21]	✓	X	X	✓	✓	✓	✓
[32]	✓	X	X	✓	✓	✓	✓
[22]	Partially	X	X	✓	Partially	Partially	Partially
[33]	X	X	✓	X	X	X	X
[31]	✓	X	X	✓	Partially	Partially	Partially
Our proposal	✓	✓	✓	✓	✓	✓	✓

In the following sections, it is presented an architectural design that combine information recorded in natural language with data collected from smart home IoT devices for nutrition monitoring.

3 Design Principles

More and more, nutrition monitoring and NLP systems are demanding IoT-enabled infrastructures in which different services can be aggregated and work together in order to produce enriched outputs after processing the information received from different sources, either speech processing services, smart devices, sensors or humans [34, 35]. On the other hand, IoT infrastructures and ecosystems need to be prepared for continuous change in response to a rapidly evolving market, as they are made up of sensors, wearable and smart devices for which there does not exist clear standardization and that are regularly replaced by new ones as technology advances [36, 37]. Therefore, the design of IoT architectures supporting NLP and nutrition monitoring needs to be approached considering the main concerns and demands of system and software architectures, paying attention to certain properties, such as flexibility and coupling between components and services [28]. Below, we describe the design principles that have guided the architecture proposal we have elaborated for nutrition monitoring systems that make use of smart devices to complement food intake records provided in natural language [27, 38, 31].

1. *Main functionalities must be easily identifiable and associated with modular building blocks.* The design of the architecture has to allow developers to unambiguously identify and direct to the main services and functionalities required for natural language processing (e.g., syntactic and semantic parsing), communication with smart devices and sensors, nutritional information extraction and data adaption and transformations according to each services specifications and interfaces. This way, developers can direct themselves to the specific services that provide the functionalities that they are seeking for.
2. *The design of the architecture must be flexible to offer full nutritional computation and recording as a service for the whole process or just for subprocesses of it.* Most of existing approaches involving NLP are conceived as monolithic, end-to-end processes which forces data streams to pass through several consecutive tasks. This fact reduces flexibility from the point of view of systems and software architects, since they cannot choose to interact with only some services and discard the ones they do not need, or to choose other proven third-party services from a different provider.
3. *Services building blocks have to exhibit clear and well-defined external interfaces and based on contracts.* By encapsulating the main functionalities for speech-based nutritional monitoring in modular building blocks behind well-defined interfaces, developers can focus on the orchestration of certain smart services of their choice. Modularity avoids tight coupling and dependence of particular implementations and fosters independent and separate evolution of different technologies and processing algorithms associated with each processing stage. The fact that the services are composed on the basis of contracts also increases their reusability and that of the components that render the services. It also contributes positively to the modifiability and extendability of systems by separating the contract (service) from the implementation (component) and avoiding side effects on other services.
4. *Ensure extendability for the incorporation of new IoT devices.* The architectures must permit to easily incorporate new smart devices that provide useful data for automatic nutritional recording. Each service must provide mechanisms to perform its functionality and to be easily adhered with the rest of devices in any system preserving encapsulation and modularity. This approach also provides more flexibility with respect to end-to-end NLP solutions so that new modules can be plugged in the architectures as technology advances and new IoT devices appear.
5. *Management and deployment of new resources on demand.* Manage the resources deployed by each of the services on demand. This way, the system has the ability to decide which services should be deployed according to the current specific needs.
6. *Understanding of input data.* In order to provide independence to each service, the input data must always be adapted to meet the requirements of each service. To provide more flexibility, these systems must implement specific services that transform and adapt the accepted input data for each service required to be invoked. Thus, each developer has the choice to support different data structures that serve as input, since internally the data will be transformed to be used appropriately by each service.
7. *Models and techniques for semantic parsing should remain simple and easily replaceable.* Similarly to IoT devices, NLP techniques are a fast changing domain. For this reason, the processing of nutritional semantics must be independent of the technique and the data structure used for semantic analysis.

4 Nutritional Monitoring Service Architecture

Next, we present our proposal for a nutritional monitoring service architecture based on natural language and smart devices for recording dietary intakes. As shown in Figure 2, our service-based architecture integrates different packages of independent services so as to offer nutritional language understanding, adjustment of nutritional records with smart home appliances and coordination elements, namely:

1. *Coordination package.* It is responsible for managing the requests, transforming the data and administrating the resources of each service.
2. *Natural Language Processing package.* It is responsible for processing the user input in natural language. As explained in Section 4.2, there are different ways in which this may be achieved. In this package we associate the processing of natural language with three services, *Syntactic Parsing*, *Entities Recognition* and *Semantic Parsing*. Each service provides resources to process the natural languages in several levels. The architecture is prepared to support all of them. This package encompasses services for parsing natural language sentences syntactic structure, recognizing the relevant entities involved and composing a semantic interpretation covering the target nutritional aspects.
3. *Adjustment of Nutritional Records with Smart Home Appliances package.* It is responsible for connecting and handling smart devices and complementing food records with data collected from them.
4. *Nutritional Info Extraction package.* It is responsible for verifying and calculating the nutritional information of all the elements from the list of entities (food, containers, etc.) derived from the user's input.
5. *SmartKitchenware Devices service.* It manages the data exchanges with smart devices. This package is responsible for implementing the drivers of the sensors and other smart devices. Each driver controls the technological requirements of the sensors that will be used in the system. This is the base to connect the smart device with the *Entities Adjustment* service, translating the data collected by the smart device into a format that is understandable for this service.

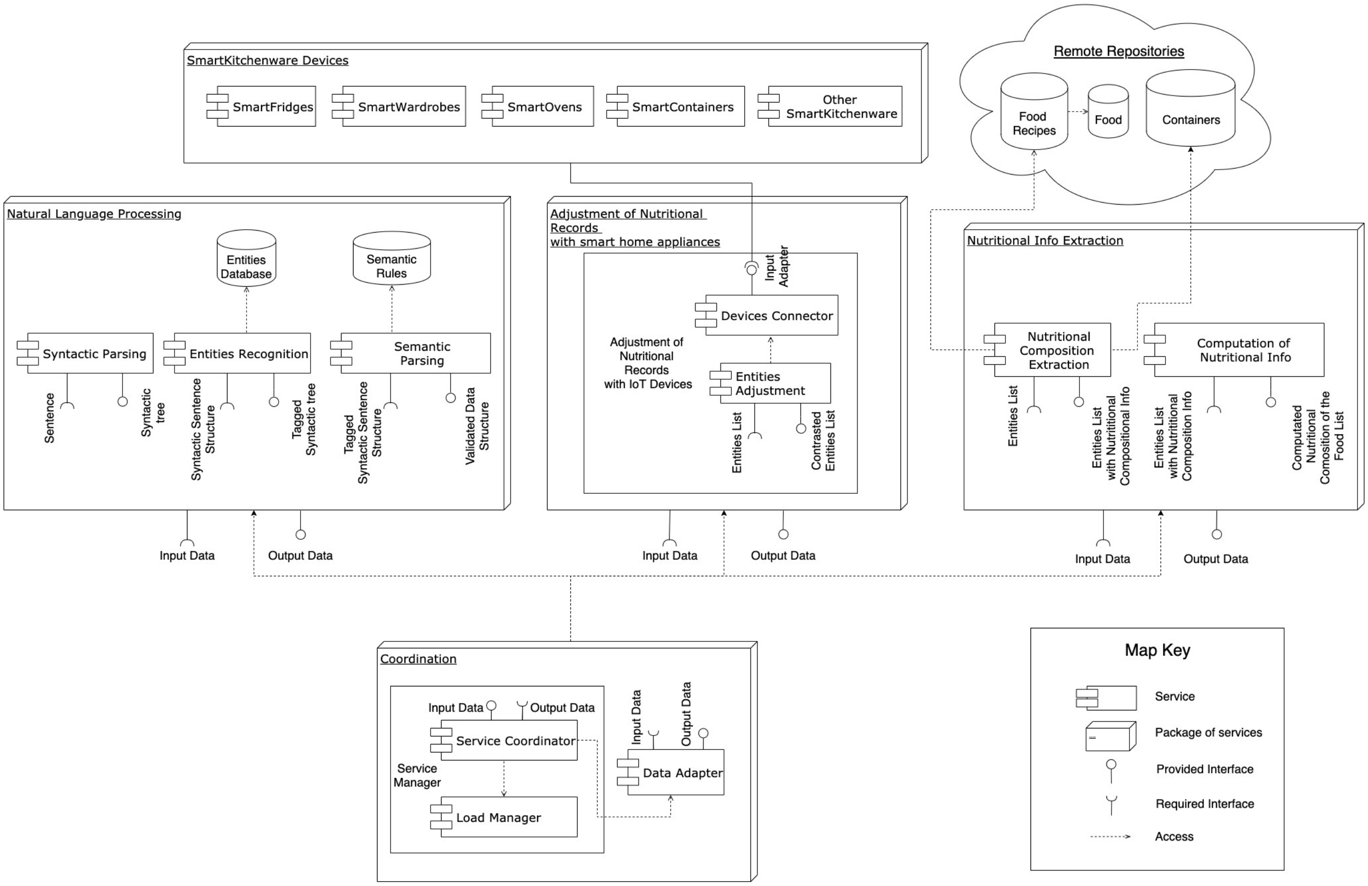


Figure 2: Service-based architecture for nutritional semantics analysis

4.1 Coordination services

The processing of natural language nutritional information follows a downstream process comprising the following stages: 1) receive a system request; 2) the request is managed adapting the input data and the functionality requested (e.g. *"calculate the nutritional composition of an intake description"*); 3) then, the required services are called performing their tasks; and 4) the result produced by the services invoked is transformed and adapted to produce the desired output data type indicated in the first request. The management of these steps is possible thanks to the *Service Coordinator*, an additional service responsible for handling the requests received.

In other words, the *Service Coordinator* analyses the request received selecting the appropriate sequence of services to be invoked in order to solve the request. Then, it decides whether the input data format is valid and calls the *"Data Adapter"* service in order to adapt the input data and execute the rest of services in the sequence. Finally the results produced by the services invoked are transformed and adapted in the specific format required in the original request. Figure 2 shows a service-based approach involving all the procedures implied in nutritional semantics. Every step in that procedure provides useful resources that could be used as part of the pipeline, in isolation or combined in different configurations in multiple contexts. For that reason, in this service architecture we have decomposed that mechanism into the following packages that contain these services:

1. *Services Coordinator:* Calculates the service sequence that should be called in order to solve the request received. This service handles and coordinates all the requests performed between the other services. For this reason, this service analyzes each request received, checking what kind of services are required in that request, and then arranges a sequence of services that will perform the process of the input data received. This service also checks the data given as an input in the request and validate if is possible to work with it in the service sequence selected. To do this the *Service Coordinator* compares the input type with the available list of data types in the *Data Adapter* service. This makes it possible to give support to as many data types as every developer may want to implement. The *Data Adapter* service transforms the data into the specific data structures required by the *Service Coordinator*. These data structures correspond with the input and output data required by every individual service proposed in this architecture. Additionally, the developers can give support to other data structures that can be compatible and transformable into the required data for these services. It should be noticed that third-party applications based on this architecture could use this coordination service or directly coordinate by themselves the interaction with the rest of services.

2. *Load Manager:* This service manages the load of each service and decides if it is necessary to deploy more resources. When the *Service Coordinator* requests a resource, this service checks the load of the required service and if is too high, the *Load Manager* deploys a new instance of the desired service depending on the available hardware resources. Additionally, this service manages all the instances running in the server, finishing all processes that are not being used in order to also save resources.

3. *Data adapter:* It receives input data and transforms them in order to produce the necessary data types for each service. In this service each developer is able to support as many input and output data types as desired.

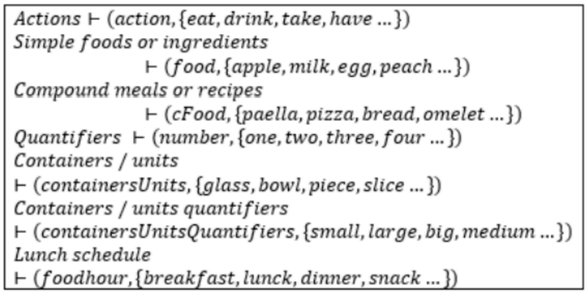
4.2 Natural Language Processing

Natural language processing (NLP) aims at understanding human language content. The first efforts were based on grammars and vocabularies that attempted to replicate human language structures, but this approach only provided good results when applied to a narrow context. In such settings, usually a keyword-based approach suffices to obtain an acceptable semantic representation. Another approach is to do a linguistic structure analysis encompassing several stages that include the identification of relevant phrases and named entities, syntactic structures and their dependencies and semantic interpretation. This approach has been the one adopted by wide-spread NLP tools such as Stanford CoreNLP [39]. At present, combinations of statistical approaches, machine-learning and language knowledge constitute the main trend [40].

In the context of nutrition, keyword spotting is very widespread. However, as discussed in our previous work [7] it may generate incomplete interpretations. This is why we proposed to perform a structure analysis in several steps comprising syntactic parsing, entities recognition and semantic analysis, which provides more accurate nutritional interpretations. Statistical analysis can be also used in this application domain, but it requires a considerable amount of training data.

Any of these three approaches can be used in the proposed architecture involving different combinations of the three main services for NLP identified, i.e., syntactic parsing, entities recognition and semantic analysis. When using keyword spotting, only the entity recognition service would be used, while with statistical analysis only the semantic parsing will be active. Note that the implementation of the services may vary in the different scenarios.

Below, we explain how these services will behave with a structured analysis, where the three main services would be active.



$Actions \vdash (action, \{eat, drink, take, have \ldots\})$
$Simple\ foods\ or\ ingredients$
$\vdash (food, \{apple, milk, egg, peach \ldots\})$
$Compound\ meals\ or\ recipes$
$\vdash (cFood, \{paella, pizza, bread, omelet \ldots\})$
$Quantifiers \vdash (number, \{one, two, three, four \ldots\})$
$Containers\ /\ units$
$\vdash (containersUnits, \{glass, bowl, piece, slice \ldots\})$
$Containers\ /\ units\ quantifiers$
$\vdash (containersUnitsQuantifiers, \{small, large, big, medium \ldots\})$
$Lunch\ schedule$
$\vdash (foodhour, \{breakfast, lunck, dinner, snack \ldots\})$

Figure 3: Entities for the semantic rules.

1. *Syntactic Parsing*: It performs a syntactic analysis of each user's input and obtains a syntactic tree with the dependencies and relationships of all elements to be interpreted. In order to do this, state-of-the-art alternatives can be used. There are numerous libraries that provide this functionality, such as the Stanford-core-NLP library [39], Apache OpenNLP [41], NLTK [42], etc. The result of the analysis is a tree where the root is the core of the sentence (the verb or action) and the other nodes represent the syntagmas and their relationships.
2. *Entities recognition*: It processes a tree produced by the *syntactic parsing* service and categorizes each node into one of the entity groups shown in Figure 3.
   (i) Actions: Words that indicate eating or drinking actions that should generate a nutritional record (e.g., *I had, I have eaten*, *I have drunk*)
   (ii) Lunch schedule: Words or verb tenses that indicate when these foods have been eaten (e.g., *breakfast*, *lunch*, *dinner*).
   (iii) Quantifiers: Words that indicate the amount (countable) of food ingested (e.g., *one*, *two*, *three*).
   (iv) Containers/units: Words that indicate containers or portions (e.g., *glass*, *bowl*).
   (v) Container/unit quantifiers: Words that indicate the sizes of the containers/units of some specific foods (e.g., *a large glass of ...*, *a small piece of...*).
   (vi) Simple food or ingredients: Words that indicate basic food (e.g., *apple*, *egg*, *water*).
   (vii) Compound meals or recipes: Words that indicate meals made up of other simple or compound foods (e.g., *omelet*, *paella*, *cake*).

   This service identifies the relevant syntagmas providing all the arguments needed for nutritional consumption interpretation. After that, it is necessary to check whether they make sense with the nutritional semantics for each entity and for other entities related to it. Finally, all the relevant syntagmas are tagged with the corresponding category (see Figure 3).
3. *Semantic Parsing*: This service interprets the meaning of the sentence taking into consideration the entities and the dependencies identified by the syntactic parser.

4.3 Adjustment of Nutritional Records with Smart Home Appliances

This service interacts with connected devices that provide information about the user's behaviour at home, collecting data that complement inputs provided by the user and which may be imprecise. In this case, it is possible to use any device that can inform about the specific food that has been consumed and/or its weight or quantity. Thus, the interaction with smart home appliances enables new ways of improving the precision of users' food intakes descriptions.

This service processes the information collected from all the devices connected and replaces the content of the *Quantifiers, Containers/units, Containers/units quantifiers and Simple foods or ingredients* among others. For example, if the user says *"I have had a bowl of cereals for breakfast"*, this phrase can be translated into: 1) Quantifiers: 1 (i.e., after *one bowl*); 2) Containers: *bowl*; 3) Simple food: *cereals*; and 4) Lunch Schedule: *breakfast*.

Without further information, the previous input would be interpreted using standard measures, i.e. replacing *"bowl"* with a standard quantity measure, such as 300mg, but this measure would not be exact. Conversely, in an IoT setting, it could be possible to obtain a more precise quantity measurement from a smart bowl and then compute a more exact estimation. In order to do it, this service would be connected to smart devices and would retrieve the measures from them. In the presented architecture, all these devices are controlled by the *Devices Connector* service which handles all the devices that are related to the processing of users descriptions. Thus, meal descriptions and associated users records about food consumption would be improved.

4.4 Nutritional Information Extraction

As far as the extraction of nutritional information is concerned, the architecture includes two services, namely:

1. Nutritional Composition Extraction: This service collects the nutritional information corresponding to a given structure of entities. Then, it provides a data structure containing the previous entities complemented with their nutritional composition extracted from databases.
2. Computing of Nutritional Info: This service processes a data structure (i.e., the *Entities List with Nutritional Composition Info* depicted in Figure 2) with information about all the entities, their nutrients and the quantities. Then, it calculates the nutritional composition of the overall structure.

5 Use case

In order to illustrate how the proposed architecture could be employed, next we describe a possible way of implementing the main services involved (Figure 4) and how a sample natural language nutritional record would be processed, step by step (Figure 5).

For every request directed to a system based on this architecture, the procedure followed is to: 1) Handle the request needs, 2) Create a plan to solve the request, 3) Adapt the data received into a different format ready to be used by the required services, 4) Manage and assign the available resources in the system, 5) Perform the service sequence in order to satisfy the request and 6) Adapt the resulting data to the data format required in the query requested on the step 1). These tasks are the responsibility of the *Service Coordinator*.

These steps are always performed regardless of the request involving the services presented in Figure 2 and illustrated in Figure 4. Firstly, the *Coordination* package manages the request received. The main service responsible for identifying and coordinating what it is required in the request is the *Service Coordinator*. This service analyzes the request received, and then, decides the services that are needed to perform that action. In order to do this, the *Service Coordinator* defines the sequence of services to be invoked, and once the sequence is determined, this service asks the *Load Manager* service to assign the deployed services that can be used. Then, the *Service Coordinator* transforms the input data received into a request call for the *Data Adapter* service. The *Data Adapter* transforms the input data into the required data structure for the next service that will be invoked by the *Service Coordinator*. After transforming the data, the *Data Adapter* returns the appropriate data structures to the *Service Coordinator*. Afterwards, the *Service Coordinator* proceeds to call the first service in the service sequence defined. The previous procedure is performed for every request received as the *Service Coordinator* calls each service in the sequence computed.

In order to clarify this process, we will describe how the input sentence *"I have had a bowl of cereals in the morning"* would be processed (see Figure 5). In order to calculate the nutritional composition of the previous sentence, the *Service Coordinator* would call the following services from the *Natural Language Processing* package (breakdown of invocations labeled as interaction number 2 in Figure 5): 2.1) *Syntactic Parsing*, 2.2) *Entities Recognition*, 2.3) *Semantic Parsing*.

Then, the *Service Coordinator* would invoke the 3) *Nutritional Composition Extraction* service from the *Nutritional Info Extraction* package, and afterwards the 4) *Entities Adjustment* from the *Adjustment of Nutritional Records with Smart Home Appliances* package.

Finally, 5) the *Computation of Nutritional Info* from the *Nutritional Info Extraction* package would obtain a calculation of different parameters of the intake, such as kilocalories, proteins, lipids, etc.

Note that our architecture is flexible and thus, it could be the case that in a particular system or platform the coordinator may decide it is only necessary to call service 2.2 (*Entities Recognition*) of the sequence above, e.g. if the sentence is to be processed only with keywords, without prior syntactic analysis and no subsequent semantic parsing. Likewise, a third-party software might request the *Service Coordinator* to only invoke the *Computation of Nutritional Information* and discard the *Adjustment of Nutritional Records with Smart Devices*. For this example, we are using the use case example presented in Figure 5. To process the sentence, the system would follow a standard coordination sequence consisting of interactions 2-5, and then would invoke the corresponding sequence of services.

In more detail, first of all, the *Service Coordinator* calls the *Syntactic Parsing service*. This service takes a sentence introduced by the user as input, analyzes it and produces a new data structure consisting of a syntactic tree including the

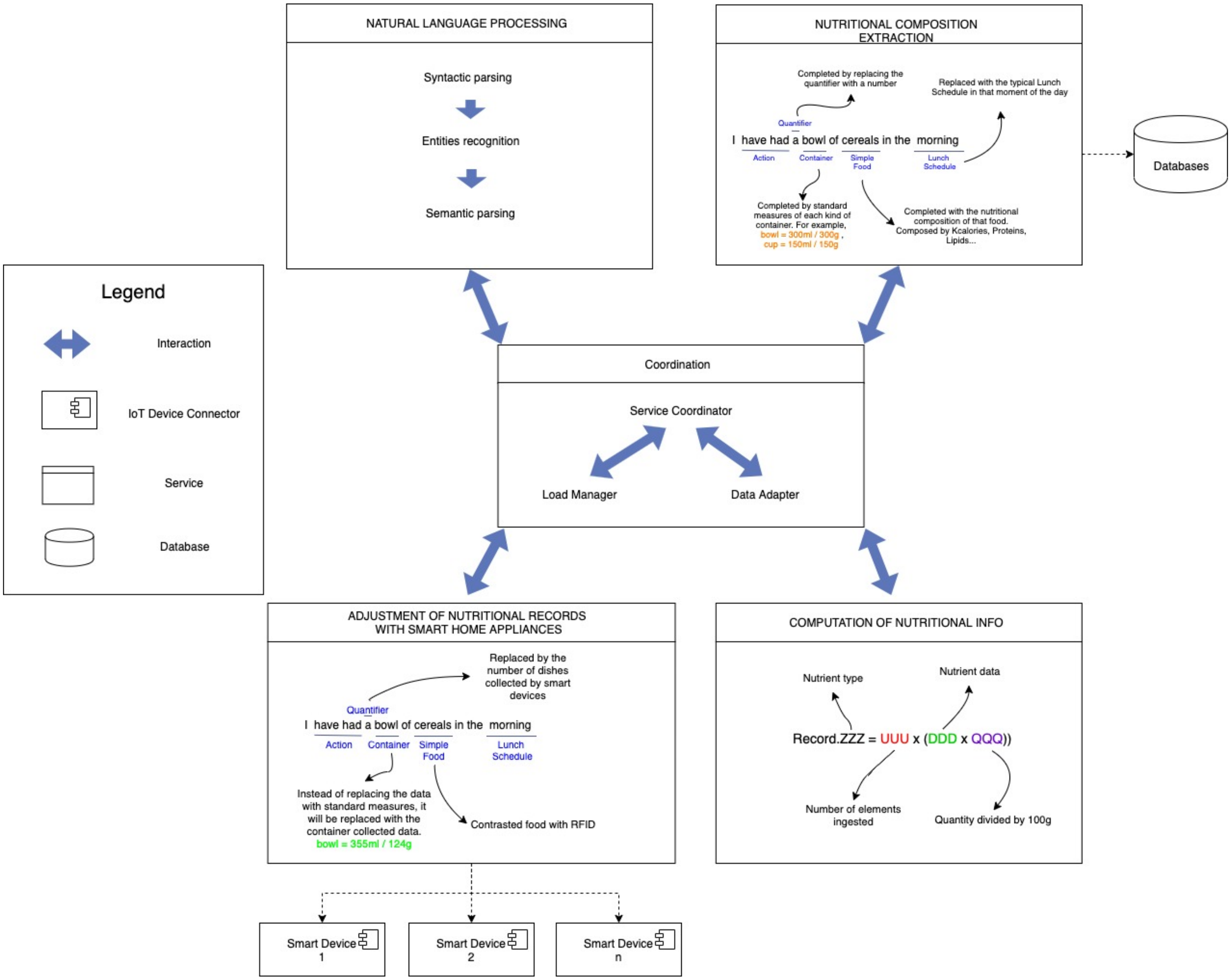
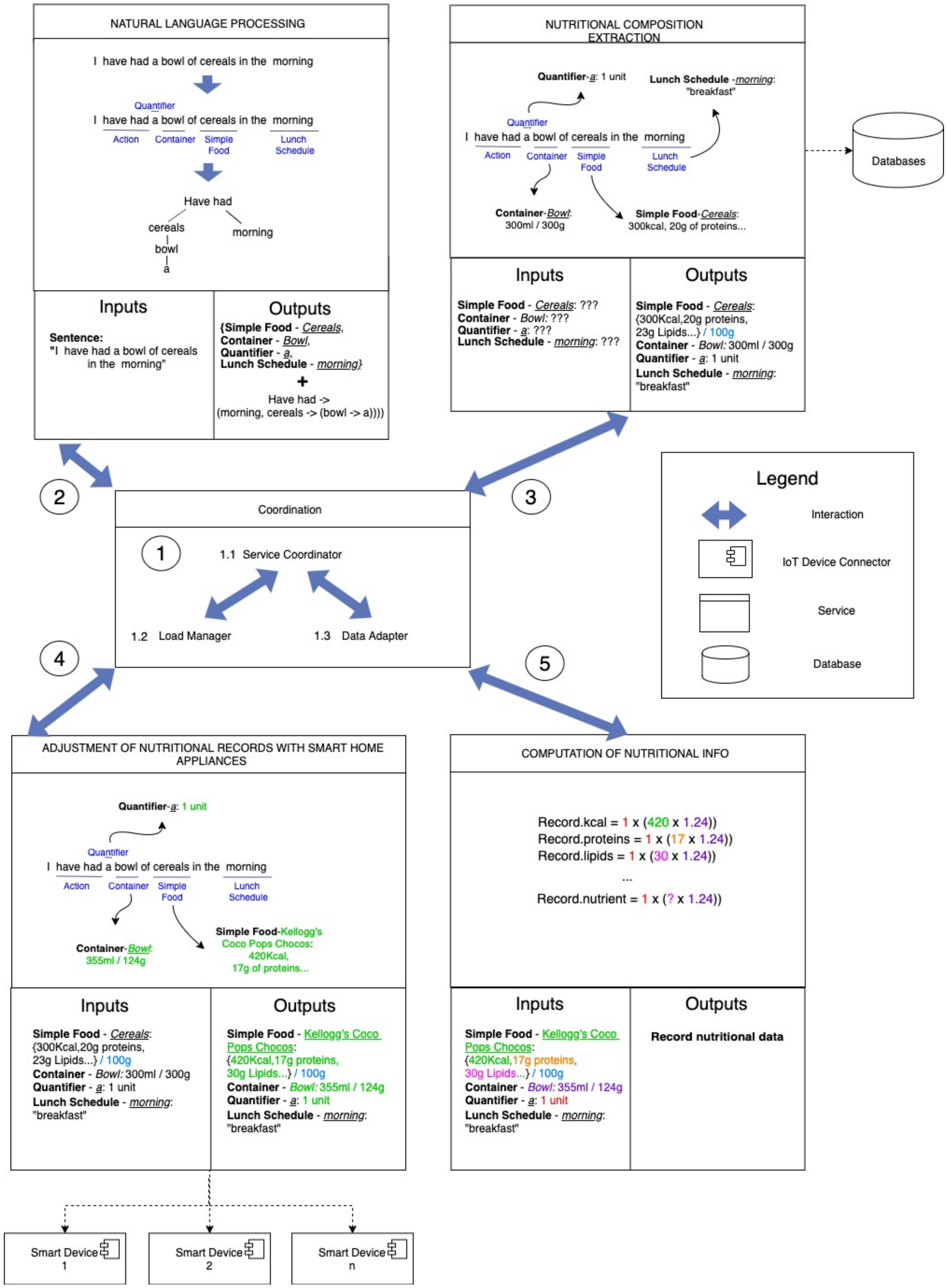

NATURAL LANGUAGE PROCESSING
Syntactic parsing
Entities recognition
Semantic parsing
NUTRITIONAL COMPOSITION EXTRACTION
Completed by replacing the quantifier with a number
Replaced with the typical Lunch Schedule in that moment of the day
Quantifier
I have had a bowl of cereals in the morning
Action
Container
Simple Food
Lunch Schedule
Completed by standard measures of each kind of container. For example, bowl = 300ml / 300g , cup = 150ml / 150g
Completed with the nutritional composition of that food. Composed by Kcalories, Proteins, Lipids...
Databases
Legend
Interaction
IoT Device Connector
Service
Database
Coordination
Service Coordinator
Load Manager
Data Adapter
ADJUSTMENT OF NUTRITIONAL RECORDS WITH SMART HOME APPLIANCES
Replaced by the number of dishes collected by smart devices
Quantifier
I have had a bowl of cereals in the morning
Action
Container
Simple Food
Lunch Schedule
Instead of replacing the data with standard measures, it will be replaced with the container collected data. bowl = 355ml / 124g
Contrasted food with RFID
COMPUTATION OF NUTRITIONAL INFO
Nutrient type
Nutrient data
Record.ZZZ = UUU x (DDD x QQQ))
Number of elements ingested
Quantity divided by 100g
Smart Device 1
Smart Device 2
Smart Device n

Figure 4: Illustration of main services and possible interactions in the proposed architecture

syntagmas identified and their syntactic function. This output data structure is received by the *Service Coordinator*, that checks whether there is another service in the service invoking sequence that is required to finish the processing. In this case, the *Service Coordinator* needs to call to the *Entities Recognition service* and the *Semantic Parsing service* consecutively. These services produce a new data structure based on the syntactic tree from the *Syntactic Parsing service*. As shown in Figure 5, the output corresponding to the sentence *"I have had a bowl of cereals in the morning"* is: 1) Quantifiers: *"a"* 1, 2) Containers: *"bowl"*, 3) Simple food: *"Cereals"* and 4) Lunch Schedule: *"morning"*.

Next, the *Service Coordinator* would call the *Nutritional Composition Extraction* service. This service would extract the nutritional composition of each entity in the semantic interpretation obtained. For example, in Figure 4 this service replaces the content of the simple food *"Cereals"* with the nutritional composition info (e.g., Kcalories, lipids, proteins, vitamins...) extracted from a connected database. Likewise, the other word groups, such as *"bowl"*, *"a"* and *"morning"* are replaced with standard measurements, like, 300ml/300g for *"bowl"*, 1 unit for *"a"* and the lunch schedule *breakfast* for *"in the morning"*. After replacing the recognized entities with the nutritional data, the *Service Coordinator* can call the *Adjustment of Nutritional Records with Smart Home Appliances* service in order to check the entities set identified in the text description using smart devices. This service checks all IoT devices connected through the *Devices Connector* that can potentially verify and complement the semantic interpretation. In our example, the previous service complemented the interpreted entities structure with information collected from the database, binding for example, *“Cereals”* with a general nutritional composition of a generic cereal product. In a similar way, *“bowl”* is replaced with a standard measure of 300ml/300g. The *Adjustment of Nutritional Records* with Smart Home Appliances services can replace some of these standard interpretations with data collected from sensors. For example, as we can see in Figure 5 *“bowl”* and *“cereals”* have been replaced with more specific measurements. This data can be recognized using smart



NATURAL LANGUAGE PROCESSING
I have had a bowl of cereals in the morning
Quantifier
I have had a bowl of cereals in the morning
Action
Container
Simple Food
Lunch Schedule
Have had
cereals
bowl
a
morning
Inputs
Sentence:
"I have had a bowl of cereals in the morning"
Outputs
{Simple Food - Cereals,
Container - Bowl,
Quantifier - a,
Lunch Schedule - morning}
+
Have had ->
(morning, cereals -> (bowl -> a)))
NUTRITIONAL COMPOSITION EXTRACTION
Quantifier-a: 1 unit
Lunch Schedule -morning: "breakfast"
Quantifier
I have had a bowl of cereals in the morning
Action
Container
Simple Food
Lunch Schedule
Container-Bowl: 300ml / 300g
Simple Food-Cereals: 300kcal, 20g of proteins...
Databases
Inputs
Simple Food - Cereals: ???
Container - Bowl: ???
Quantifier - a: ???
Lunch Schedule - morning: ???
Outputs
Simple Food - Cereals:
{300Kcal,20g proteins,
23g Lipids...} / 100g
Container - Bowl: 300ml / 300g
Quantifier - a: 1 unit
Lunch Schedule - morning:
"breakfast"
2
3
Coordination
1
1.1 Service Coordinator
1.2 Load Manager
1.3 Data Adapter
4
5
Legend
Interaction
IoT Device Connector
Service
Database
ADJUSTMENT OF NUTRITIONAL RECORDS WITH SMART HOME APPLIANCES
Quantifier-a: 1 unit
Quantifier
I have had a bowl of cereals in the morning
Action
Container
Simple Food
Lunch Schedule
Container-Bowl: 355ml / 124g
Simple Food-Kellogg's Coco Pops Chocos: 420Kcal, 17g of proteins...
Inputs
Simple Food - Cereals:
{300Kcal,20g proteins,
23g Lipids...} / 100g
Container - Bowl: 300ml / 300g
Quantifier - a: 1 unit
Lunch Schedule - morning:
"breakfast"
Outputs
Simple Food - Kellogg's Coco Pops Chocos:
{420Kcal,17g proteins,
30g Lipids...} / 100g
Container - Bowl: 355ml / 124g
Quantifier - a: 1 unit
Lunch Schedule - morning:
"breakfast"
COMPUTATION OF NUTRITIONAL INFO
Record.kcal = 1 x (420 x 1.24))
Record.proteins = 1 x (17 x 1.24))
Record.lipids = 1 x (30 x 1.24))
...
Record.nutrient = 1 x (? x 1.24))
Inputs
Simple Food - Kellogg's Coco Pops Chocos:
{420Kcal,17g proteins,
30g Lipids...} / 100g
Container - Bowl: 355ml / 124g
Quantifier - a: 1 unit
Lunch Schedule - morning:
"breakfast"
Outputs
Record nutritional data
Smart Device 1
Smart Device 2
Smart Device n

Figure 5: Illustration of the processing of a natural language sentence

devices such as smart dishes that recognize the volume/weight and smart fridges or containers that are aware of the elements that they contain. After this, the *Service Coordinator* performs the last service call in the sequence, invoking the *Computation of Nutritional Info*. All the data completed in the previous steps is used in this service, producing the nutritional calculus of all the recognized entities.

Although the previous example depicts a sequential execution, the proposed architecture provides modularity, loose coupling and independence between services. Thus, each service just requires a data structure to produce its output and this input structure can be provided in different data formats. To do this, developers can implement a *Data Adapter* in order to support as many data formats as required by the particular services implemented.

6 Conclusions and Future Work

Nutrition mobile applications are becoming a promising tool for users to become aware of their diet and manage their health, as well as for facilitating their remote monitoring by health professionals. Most of these applications consist of food frequency questionnaires to control dietary habits. The inclusion of natural language interfaces in this kind of applications provides users with more flexibility to express dietary intakes in their own words and in more detail than through traditional text-based interfaces. In addition, smart home appliances, devices and sensors connected to the Internet (directly or through proxy devices) in IoT settings make it possible to complement the information provided by the users with other relevant pieces of information and more precise measurements gathered from the environment, e.g. from smart kitchenware and appliances.

Although intertwining IoT and natural language processing can bring many benefits for nutritional applications, it also presents numerous challenges. In this paper we have discussed the current requirements and challenges for developing systems based on services that use smart home devices connected to the Internet for nutritional semantics computation. In particular, we have pointed out the lack of models and architectures that ensure the encapsulation of relevant services, the need for independence of the technology used by each of them and the required ease of management and deployment of new resources to make applications extendable, specially in IoT settings as new devices and applications enter the scene.

To address this important challenge, we have proposed and illustrated a service-oriented architecture for dietary intake monitoring comprising a set of independent services based on natural language processing and connected smart home appliances. Our proposal makes it possible to employ multiple computing resources and data formats as the encapsulation of the services presented fosters the independence of the technology used. This architecture also allows developers to extend the functionalities described in this paper by adding new services easily. Moreover, the presented architecture has been conceived as a set of services for nutrition monitoring purposes. On top of it, multi-faceted platforms could be built addressing, not only monitoring the nutritional dimension, but also others related to wellness and health, such as activity habits, gait, posture, etc. The combination of all these dimensions together with appropriate user modelling and profiling would deserve further work and research.

This work is a step forward towards the ideal of mobile, ubiquitous, transparent and accurate self-reporting of dietary habits and nutrients intake. It sets the basis for prospective research about how IoT settings can be a useful tool in the nutrition analysis and coaching domain. For future work we plan to evaluate the proposed architecture and services in terms of reliability, availability, conceptual integrity and other key aspects identified in previous work. In particular, the kind of architecture evaluation that we plan carry out will follow standard practice in this kind of research ([43, 44, 45] and based on [46] guidelines, including questionnaires and animated multimedia of software architecture models to be evaluated by world-wide experts on systems and software architectures and experienced developers. The huge challenge of combining multiple, possible conflicting or contradictory, sources of information (wearable body sensors, computer vision, nutritional databases, smart devices, frequency food diaries, etc.), in order to determine nutrients intake with precision still remains.

Acknowledgements

This research has been supported by the project DEP2015-70980-R of the Spanish Ministry of Economy and Competitiveness (MINECO) and European Regional Development Fund (ERDF), the European Union's Horizon 2020 research and innovation programme under grant agreement No 823907, as well as, received inputs from the COST Action IC1303 AAPELE.

References

[1] Kristen MJ Azar, Lenard I Lesser, Brian Y Laing, Janna Stephens, Magi S Aurora, Lora E Burke, and Latha P Palaniappan. Mobile applications for weight management: theory-based content analysis. *American journal of preventive medicine*, 45(5):583–589, 2013.

[2] Cathaleene Macias, Trishan Panch, Yale M Hicks, Jason S Scolnick, David Lyle Weene, Dost Öngür, and Bruce M Cohen. Using smartphone apps to promote psychiatric and physical well-being. *Psychiatric Quarterly*, 86(4):505–519, 2015.

[3] Ángel Ruiz-Zafra, Kawtar Benghazi, Manuel Noguera, and José Luis Garrido. Zappa: An open mobile platform to build cloud-based m-health systems. In *Ambient intelligence-software and applications*, pages 87–94. Springer, 2013.

[4] Enrique Moguel, Javier Berrocal, Juan M. Murillo, José Garcia-Alonso, David Mendes, Cesar Fonseca, and Manuel Lopes. Monitoring food intake in an aging population: A survey on technological solutions. *Proceedings*, 2(19), 2018.

[5] Ronilda Lacson and William Long. Natural language processing of spoken diet records (sdrs). In *AMIA Annual Symposium Proceedings*, volume 2006, page 454. American Medical Informatics Association, 2006.

[6] N. Hezarjaribi, S. Mazrouee, and H. Ghasemzadeh. Speech2health: A mobile framework for monitoring dietary composition from spoken data. *IEEE Journal of Biomedical and Health Informatics*, 22(1):252–264, Jan 2018.

[7] Antonio Benítez-Guijarro, Zoraida Callejas, Manuel Noguera, and Kawtar Benghazi. Introducing computational semantics for natural language understanding in conversational nutrition coaches for healthy eating. In *Multidisciplinary Digital Publishing Institute Proceedings*, volume 2, page 506, 2018.

[8] Anna M. Rangan, Laurissa Tieleman, Jimmy C. Y. Louie, Lie Ming Tang, Lana Hebden, Rajshri Roy, Judy Kay, and Margaret Allman-Farinelli. Electronic dietary intake assessment (e-dia): relative validity of a mobile phone application to measure intake of food groups. *British Journal of Nutrition*, 115(12):2219–2226, 2016.

[9] Arthur A Stone, Saul Shiffman, Joseph E Schwartz, Joan E Broderick, and Michael R Hufford. Patient compliance with paper and electronic diaries. *Controlled Clinical Trials*, 24(2):182 – 199, 2003.

[10] Julia K Kolodziejczyk, Gina Merchant, and Gregory J Norman. Reliability and validity of child/adolescent food frequency questionnaires that assess foods and/or food groups. *Journal of pediatric gastroenterology and nutrition*, 55(1):4–13, 2012.

[11] Peter Leijdekkers, Valerie Gay, and Elaine Lawrence. Smart homecare system for health tele-monitoring. In *First International Conference on the Digital Society (ICDS'07)*, pages 3–3. IEEE, 2007.

[12] Hanshen Gu and Dong Wang. A content-aware fridge based on rfid in smart home for home-healthcare. In *2009 11th International Conference on Advanced Communication Technology*, volume 2, pages 987–990. IEEE, 2009.

[13] Marketa Dolejsova. Edible speculations in the parlour of food futures. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, page alt13. ACM, 2018.

[14] Anjishnu Kumar, Arpit Gupta, Julian Chan, Sam Tucker, Björn Hoffmeister, and Markus Dreyer. Just ASK: building an architecture for extensible self-service spoken language understanding. *CoRR*, abs/1711.00549, 2017.

[15] Oyungerel Byambasuren, Sharon Sanders, Elaine Beller, and Paul Glasziou. Prescribable mHealth apps identified from an overview of systematic reviews. *npj Digital Medicine*, 1:2398–6352, 2018.

[16] Juliana Chen, Luke Gemming, Rhona Hanning, and Margaret Allman-Farinelli. Smartphone apps and the nutrition care process: Current perspectives and future considerations. *Patient Education and Counseling*, 101(4):750 – 757, 2018.

[17] Zhiping Yu, Claudia Sealey-Potts, and Judith Rodriguez. Dietary self-monitoring in weight management: Current evidence on efficacy and adherence. *Journal of the Academy of Nutrition and Dietetics*, 115(12):1931 – 1938, 2015.

[18] Joao Luis Zeni Montenegro, Cristiano André da Costa, and Rodrigo da Rosa. Survey of conversational agents in health. *Expert Systems with Applications*, 129:56 – 67, 2019.

[19] Jessica R.L. Lieffers, Jose F. Arocha, Kelly Grindrod, and Rhona M. Hanning. Experiences and perceptions of adults accessing publicly available nutrition behavior-change mobile apps for weight management. *Journal of the Academy of Nutrition and Dietetics*, 118(2):229 – 239.e3, 2018.

[20] Verena Fuchsberger. Ambient assisted living: elderly people's needs and how to face them. In *Proceedings of the 1st ACM international workshop on Semantic ambient media experiences*, pages 21–24. ACM, 2008.

[21] Rubén Blasco, Álvaro Marco, Roberto Casas, Diego Cirujano, and Richard Picking. A smart kitchen for ambient assisted living. *Sensors*, 14(1):1629–1653, 2014.

[22] Rodrigo Souza, João Lopes, Cláudio Geyer, Anderson Cardozo, Adenauer Yamin, and Jorge Barbosa. An architecture for iot management targeted to context awareness of ubiquitous applications. 24(10):1452–1471, 2018.

[23] Wolfgang Minker, Ramón López-Cózar, and Michael Mctear. The role of spoken language dialogue interaction in intelligent environments. *Journal of Ambient Intelligence and Smart Environments*, 1(1):31–36, 2009.

[24] Mauro Dragoni, Tania Bailoni, Claudio Eccher, Marco Guerini, and Rosa Maimone. A semantic-enabled platform for supporting healthy lifestyles. In *Proceedings of the Symposium on Applied Computing*, SAC '17, pages 315–322, New York, NY, USA, 2017. ACM.

[25] Jakob Axelsson and Mats Skoglund. Quality assurance in software ecosystems: A systematic literature mapping and research agenda. *Journal of Systems and Software*, 114:69 – 81, 2016.

[26] Seng Loke. *Context-aware pervasive systems: architectures for a new breed of applications*. Auerbach Publications, 2006.

[27] Martin Becker. Software architecture trends and promising technology for ambient assisted living systems. In *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2008.

[28] S. F. Ochoa and F. J. Gutierrez. Architecting e-coaching systems: A first step for dealing with their intrinsic design complexity. *Computer*, 51(3):16–23, March 2018.

[29] Cristina Roda, Elena Navarro, Uwe Zdun, Víctor López-Jaquero, and Georg Simhandl. Past and future of software architectures for context-aware systems: A systematic mapping study. *Journal of Systems and Software*, 146:310 – 355, 2018.

[30] Idowu Ayoola, Mart Wetzels, Peter Peters, Sander van Berlo, and Loe Feijs. Do change platform: A service-based architecture for secure aggregation and distribution of health and wellbeing data. *International Journal of Medical Informatics*, 117:103 – 111, 2018.

[31] Sarada Prasad Gochhayat, Pallavi Kaliyar, Mauro Conti, Prayag Tiwari, V.B.S. Prasath, Deepak Gupta, and Ashish Khanna. Lisa: Lightweight context-aware iot service architecture. *Journal of Cleaner Production*, 212:1345 – 1356, 2019.

[32] Rita Zgheib, Emmanuel Conchon, and Rémi Bastide. Engineering iot healthcare applications: Towards a semantic data driven sustainable architecture. In Kostas Giokas, Laszlo Bokor, and Frank Hopfgartner, editors, *eHealth 360°*, pages 407–418, Cham, 2017. Springer International Publishing.

[33] C. Chen, M. Karvela, M. Sohbati, T. Shinawatra, and C. Toumazou. Person—personalized expert recommendation system for optimized nutrition. *IEEE Transactions on Biomedical Circuits and Systems*, 12(1):151–160, Feb 2018.

[34] Ángel Ruiz-Zafra, Kawtar Benghazi, Constandinos X. Mavromoustakis, and Manuel Noguera. An iot-aware architectural model for smart habitats. In *16th IEEE International Conference on Embedded and Ubiquitous Computing, EUC 2018, Bucharest, Romania, October 29-31, 2018*, pages 103–110, 2018.

[35] Afrah Djeddar, Hakim Bendjenna, Abdelkrim Amirat, Philippe Roose, and Lawrence Chung. An emerging multi-paradigm for representing mobile applications' architectures using heterogeneous conceptual bricks. *IJCAT*, 57(1):1–13, 2018.

[36] Ángel Ruiz-Zafra, Manuel Noguera, Kawtar Benghazi, and Sergio F. Ochoa. A model-driven approach for wearable systems developments. *International Journal of Distributed Sensor Networks*, 11(10):637130, 2015.

[37] Ahyoung Choi and Hangsik Shin. Longitudinal healthcare data management platform of healthcare iot devices for personalized services. 24(9):1153–1169, 2018.

[38] Chi-Kun Eddy Lin, Ting-Hui Wang, and Jar-Ferr Kevin Yang. Touch doctor—a nutrition control service system developed under living lab methodology. *International Journal of Automation and Smart Technology*, 2(3):253–263, 2012.

[39] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60, 2014.

[40] Julia Hirschberg and Christopher D. Manning. Advances in natural language processing. *Science*, 349(6245):261–266, 2015.

[41] Kurt Hornik. opennlp: Apache opennlp tools interface, 2014. *R package version 0.2-3*, 2014.

[42] Edward Loper and Steven Bird. Nltk: the natural language toolkit. *arXiv preprint cs/0205028*, 2002.

[43] Vladimir Villarreal, Jesus Fontecha, Ramon Hervas, and Jose Bravo. Mobile and ubiquitous architecture for the medical control of chronic diseases through the use of intelligent devices: Using the architecture for patients with diabetes. *Future generation computer systems*, 34:161–175, 2014.

[44] Liming Zhu, Aybüke Aurum, Ian Gorton, and Ross Jeffery. Tradeoff and sensitivity analysis in software architecture evaluation using analytic hierarchy process. *Software Quality Journal*, 13(4):357–375, 2005.

[45] Antonio Benítez-Guijarro, Ángel Ruiz-Zafra, Zoraida Callejas, Nuria Medina-Medina, Kawtar Benghazi, and Manuel Noguera. General architecture for development of virtual coaches for healthy habits monitoring and encouragement. *Sensors*, 19(1):108, 2019.

[46] Roy T Fielding and Richard N Taylor. *Architectural styles and the design of network-based software architectures*, volume 7. University of California, Irvine Doctoral dissertation, 2000.