A Flexible Knowledge-Based Architecture for Supporting the Adoption of Healthy Lifestyles with Persuasive Dialogs

Mauro Dragoni, Tania Bailoni, Rosa Maimone, Michele Marchesoni, and Claudio Eccher

1 Introduction

Recent studies like [1] and [2] have shown that living a long and healthy life prevents cognitive decline, obesity, disability, and death from major chronic diseases (like diabetes, cardiovascular disease, and several forms of cancer). In the domain of health and well-being, the use of information and communication technology (ICT)-based motivational systems that produce user-tailored messages can be effective tools to persuade and motivate people to change their behavior toward a healthier lifestyle (adopting and maintaining correct diet and active living); the user-tailored messages can be generated by reasoning on data gathered from the user, using his/her personal devices and off-the-shelf wearable sensors [3].

However, engaging people in developing and maintaining healthier patterns of living is a challenging task. To this end, generating effective personalized recommendations implies, for example, the justification of given suggestions and the adaptation of messages in response to the modification of the environment and of the user status. For this reason, as opposed to hardwired persuasive features, systems that apply general reasoning capabilities to provide flexible persuasive communication based on rich and diverse linguistic outputs are required. In this context, modeling persuasion mechanisms and performing flexible and context-dependent persuasive actions are more ambitious than most current approaches on persuasive technologies (see *Captology* [4]). In fact, the design of a flexible system,

M. Dragoni (✉) · T. Bailoni · R. Maimone · M. Marchesoni · C. Eccher
Fondazione Bruno Kessler, Trento, Italy
e-mail: dragoni@fbk.eu; tbailoni@fbk.eu; rmaimone@fbk.eu; marchesoni@fbk.eu; cleccher@fbk.eu

S. Consoli et al. (eds.), *Data Science for Healthcare*,
https://doi.org/10.1007/978-3-030-05249-2_9

applicable to different domains, poses relevant challenges related to the implemented persuasive strategies and the architecture that must ensure the independence of the information processing machinery from both the domain of application and the language in which messages are generated.

In this chapter, we present a motivational platform for supporting the monitoring of users' behaviors and for persuading them to follow healthy lifestyles. The contribution of this chapter extends the work presented in [5–8]. The aim of our research is to develop a general purpose persuasive architecture flexible and easily portable to different domains of application and adaptable to new languages. To this end, we first had to individuate the components that are domain and language independent and those that are specific of a domain and language, so to reduce as much as possible portability efforts. Then, we had to rely on the use of knowledge referring to the different domains of discourse: like knowledge on food content and nutrients, categorization and effort of physical activities, user attitudes and preferences, linguistic knowledge, and environment information (places, weather, etc.).

Semantic technologies are used for modeling all relevant information and for fostering reasoning activities by combining user-generated data and domain knowledge. Moreover, the integrated ontology supports the creation of the dynamic interfaces used by domain experts for designing monitoring rules. Contextually, our system aims at inducing the user to follow specific behaviors and to maintain them over a certain timespan. The system takes into account the "social environment," exploits the situational context, and enhances emotional aspects of communication. In this scenario, what really matters is not simply the content but the overall impact of the communication. In order to validate the proposed architecture, we developed mobile applications that a group of 119 users adopted for 7 weeks. Our aim was to observe if the use of our platform would be able to support them in improving the quality of their lifestyle.

2 Related Work

In recent years, persuasive technologies have been applied in multiple areas of research. Healthcare is one of the most investigated fields, not least because it takes advantage of the spread of ICT devices. In the literature, there are many studies regarding health promotion and disease risk prevention, which address system design and implementation. In general, these works can be developed using two approaches: *vertical* and *horizontal*.

Many of the study published regard *vertical* approaches; these systems are tailored for a specific domain and usually rely on ad hoc solutions such as canned texts. These systems have the advantage of being effective on the domain, but their flexibility is usually low, and an extensive reengineering is required to port them to new domains.

On the contrary, *horizontal* approaches are not **bound** to a particular domain, and they try to address the problem of rich persuasive generation from a general perspective; they have potential of being easily portable and adaptable but usually remain at a theoretical or proof of concept level. In [9–11] and [12], the authors give an important contribution defining a persuasive systems design model for behavior change support systems; these works detail the concepts and methodology for the design and evaluation of flexible persuasive behavior change systems. Focusing on generative aspects, some seminal works on argumentation-based text generation have been proposed [13, 14], but the authors focus on the validity of the generated messages rather than their effectiveness. A more recent approach, presented in [15], introduces a persuasion framework that combines generation with information gathered from social media. In general, a thorough review and classification of available persuasive natural language generation (NLG) *horizontal* systems can be found in [16].

Turning to the specific task of generating motivational messages for health promotion, in [17] the authors present a theoretical framework for representing real-time tailored messages in behavior change applications that can be adapted to different generation strategies ranging from canned text to deep generation. Four important properties of a motivational message are considered: timing, intention, content, and representation. This framework inspired the development of the persuasive engine integrated into our platform. However, differently from our work, it has not been instantiated in any real system.

The following studies based on *vertical* approaches give an objective validation on the use of tailored and personalized persuasive messages in behavior change. In [18], the authors present a systematic review of mobile phone and web-based text messages (reminders, information provision, tailored and standardized supportive messages, and self-monitoring instructions) to promote mental health. Considering 36 studies, 35 of them show the positive impact of text messaging on patient motivation to improve their health and encourage treatment. Other studies, such as [19] and [20], show that tailored and personalized messages with variety in frequency are most effective, mainly in physical activity and smoke cessation interventions. In [21], researchers conducted an exploratory study to evaluate the tailored text messaging acceptability when used in the maintenance phase (i.e., the phase where users already follow healthy lifestyles and they have to preserve them). Women involved in the study received encouragement messages to adopt healthy behavior and text messages to prompt self-reported weight, goal setting, and goal monitoring. Also in this case, positive results show the importance of the tailored content and scheduling of text messages. Finally, in [22] the authors investigate about the use of well-being recommendation strategies on workplace. Our platform improves the dynamic creation of the persuasive messages, which are based on the profile of the specific user and the data he/she inputs.

To the best of our knowledge, there is no work that merges in a systematic way both *horizontal* and *vertical* approaches, and our work is the first attempt in this direction.

3 The Requirements for Being Effectively Persuasive

To obtain an effective behavioral change, a persuasive system should meet several requirements. Based on the analysis of the framework proposed in [17] combined with the scenarios we want to address, the following requirements were identified:

- *Sense* and *reason* on the actual context of the interaction: so to be able to decide whether to intervene or not given the current circumstances (e.g., avoid sending messages during a meeting).
- Use different strategies connected to the intended outcome (*pre*, *post*, or *during*):
  - *Pre* strategies are meant to be used before an action takes place, and it is forecast to happen in a short period of time (e.g., lunch). These strategies are meant to drive the user into a desired behavior or to divert him/her from an unwanted one.
  - *During* strategies are meant to be used when a prolonged action is taking place (e.g., working out) to support or modify it (e.g., *keep on, there are only 100 steps left*, or *slow down, you are walking too fast*).
  - *Post* strategies are meant to be used after an action took place as a reinforcement feedback or negative feedback in view of future actions of the same kind (e.g., if a user ate too much meat).They can also be used to induce a *compensatory* action [23].
- Choose the proper timing for its intervention so to maximize the likelihood to obtain the desired effect (e.g., a message aiming at convincing the user to walk home after work is more effective if sent right before the user leaves the office rather than when he/she arrives in the morning).
- Use several persuasive techniques/strategies so to choose the most appropriate in a given situation (e.g., the mood of the user can drive the selection of the available strategies, or the history of the interactions can block the repetition of arguments already used in favor of new ones).
- *Plan* complex messages and *produce* rich and natural linguistic outputs.

In general, language is the key mean for persuasion since it is the medium that allows for a more versatile and richer expression of arguments for convincing users to adopt the desired behavior. Virtually any persuasive strategy can be realized linguistically, while this is not true for other media. Then, an additional challenge is mapping persuasion strategies to linguistic realization suitable for the domain of interest.

4 Technological Challenges for Building a Flexible System

The challenges presented above for designing systems supporting an effective behavior change call for a careful design and planning of strategies to be used. A technological architecture has to support effectively their integration and use, using diverse technologies and applications.

Figure 1 shows the diagram we propose for the realization of this kind of platform. The diagram relies on four (4) layers:

- the *Input Layer* is responsible for receiving data from users or sensors, through explicit input or by event detection.
- the *Knowledge Layer*, called HeLiS, contains (1) the structured knowledge linking provided data with domain information and (2) the reasoner used for elaborating such data.
- the *Persuasive Layer*, called PersEO, contains the linguistic strategies and vocabularies for generating the feedback sent to users.
- the *Output Layer* is in charge of presenting the feedback to users.

In this section, we provide a brief introduction to all layers by highlighting which are the main challenges they have to address. A focus on the *Knowledge Layer* and on the *Dialog-Based Persuasive Layer* is provided in Sects. 5 and 6, respectively.

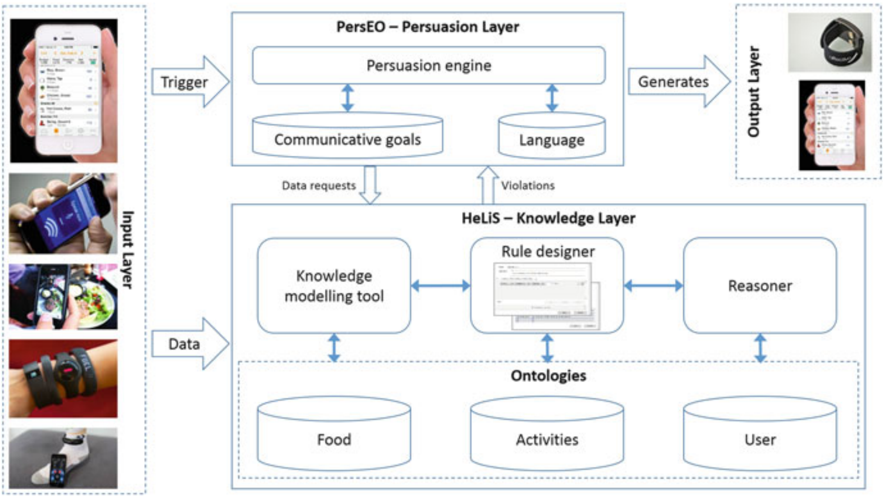

Fig. 1 The schema of the proposed platform architecture

4.1 Input Layer

The Input Layer is responsible for detecting events triggering the platform activities and accounts for the ability of a persuasive system of sensing the context of interaction. These events can be of two types: (1) data input, where data are sent from the Input Layer to the Knowledge Layer (presented in Sect. 4.2), and (2) context communication, where contextual information is sent from the Input Layer to the Persuasion Layer (presented in Sect. 4.3) that may exploit this information for persuasive purposes.

Here the distinction between data input and context communications relies in the use of parameters by the system. Input data represent facts of the world related to the user's behavior that trigger Knowledge Layer rules in the specific domain (e.g., the assumption of meals encouraged or discouraged by the Mediterranean diet recommendations). Context communication is related to the environment in which the user is acting (e.g., timing or localization) and provides information to the *Dialog-Based Persuasion layer* allowing the choice of the most appropriate message generation strategy. For example, assuming to have the required knowledge and network support, an example of exploitable context information is the localization of a user in front of a vending machine at midmorning. Based on the history of past violations, the system could suggest avoiding specific foods, for example, packaged snacks.

The Input Layer includes the possibility of both, using human computer-based solutions, like mobile applications, and connecting the platform to wearable devices or external infrastructures (e.g., the city bus stop map or the weather forecasts) enabling the automatic data transfer to the platform. One of the most prominent challenges in the design of an effective and efficient Input Layer is to reduce as much as possible the time-consuming activities on the user side. Indeed, when we refer to the digital health domain and, in particular, when we consider the nutrition and activity dimensions, the effort necessary for providing all information required by the whole platform might be significantly time-consuming (i.e., the input of all consumed foods).

4.2 Knowledge Layer

To support natural argumentation and (emotional) persuasion and to allow reasoning on the possible arguments to be put forward, it is necessary to define new methods for representing knowledge, for reasoning on it, and for generating natural language and multimodal messages (both in monological and dialogical situations). All these aspects are primarily driven by *persuasion* reasons rather than ontological ones.

Based on this consideration, we propose a *Knowledge Layer* encompassing two kinds of information:

- Augmented Domain Knowledge: the structured representation of the domain of interest including those relations that are relevant for persuasion purposes, such as the similar-taste relation or the categorization of food properties into negative and positive ones. In general, it is necessary to model all the concepts supporting the behavior change purpose and the relationships between them. These concepts will furnish the basis for the *arguments* included in the feedback provided to users.
- Monitoring Knowledge: the structured representation of the rules driving the behavior change process. Here, it is necessary to define which aspects of users' behaviors have to be monitored by the platform in order to produce proper feedbacks.

4.3 Persuasion Layer

In modeling the *Persuasion Layer*, we tried to address the overall challenging structure for building effective natural language generation (NLG) persuasive systems. In particular we expanded on the idea presented in [24] of a classification of basic persuasive strategies (what to say, how to say), supporting strategies (i.e., strategies that are meant to give support to a specific claim), and a meta-reasoning model that works on these strategies (selection and ordering of basic strategies). This model is built by taking into consideration studies coming both from social psychology and philosophy and from the area of natural argumentation. The model is neither domain nor language specific and it eases the portability of systems that are based on it.

The role of the *Persuasion Layer* is not limited to the generation of single messages. Indeed, the application of a persuasion strategy generally requires more than one interaction with the user. Thus, the *Persuasion Layer* is also in charge of managing the relationships between single messages and understanding information provided by users in order to build a reasonable conversation with the user.

4.4 Output Layer

The last layer, the *Output Layer*, is in charge of closing the loop by providing the feedback to users. It is represented by one of the many devices able to receive the data produced by the *Persuasion Layer* and to convey the physical feedback to users (textual or audio messages, graphics alerts, vibration, etc.)

The main challenge this layer has to address is to find the best trade-off between two dimensions:

- Type of feedback: it is necessary to determine the optimal way for communicating with users. This choice is strongly associated with the kind of device used for providing the feedback.
- Presentation: how content generated by the Persuasion Layer is presented to users is relevant for completing the process of supporting the behavior change.

The *Output Layer* is responsible of designing an effective presentation strategy based on the hardware capabilities of the device used. Finally, the output provided by the platform could also be a further request of inputs; thus, a connection between the two layers has to be foreseen.

5 HeLiS: The Knowledge Layer

We presented in Sect. 4.2 the challenges related to the design of an effective Knowledge Layer including (1) the modeling of an augmented domain ontology containing specific concepts for supporting the monitoring activity, (2) the implementation of a tool for supporting the work of domain experts, and (3) the realization of a reasoning mechanism enabling the semantic analysis of the data input to the system.

Here, we provide further details about the Knowledge Layer integrated within our platform. We provide an overview of the ontology branches describing the monitoring rules associated with users (or profiles), the concepts that are instantiated for storing data, and the concepts modeling detected violations. Then, we show how the platform supports the domain experts in defining monitoring rules. Finally, we describe how reasoning is implemented to evaluate the rules.

These three components allow to cope with the technological challenges concerning the realization of a Knowledge Layer capable of providing a knowledge artifact able to support the storage of user data by adopting a well-defined conceptual model and to perform reasoning operation on them in order to enable the generation of contextual message by the platform. Moreover, the development of software facilities dedicated to the domain experts allows to make the overall reasoning and message generation processes more flexible with respect to the context.

5.1 *The Augmented HeLiS Ontology*

The concepts of the HeLiS ontology of main interest for this chapter are shown in Fig. 2 and are organized in four main branches: (1) food, (2) activity, (3) monitoring, and (4) user. Further details about the ontology are provided in [25, 26] and online.[1]

[1] http://w3id.org/helis.

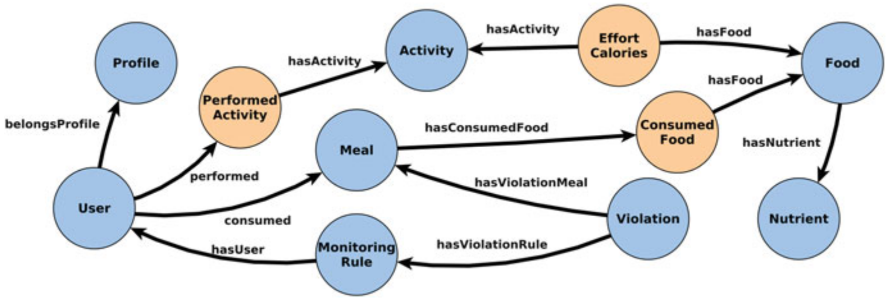

Fig. 2 The HeLiS ontology

The **food** branch is responsible for modeling the instances macro-grouped under the `BasicFood` (986 instances) and `Recipe` (4408 instances) concepts. Instances of the `BasicFood` concept describe foods for which micro-information concerning `Nutrients` (carbohydrates, lipids, proteins, minerals, and vitamins) is available, while instances of the `Recipe` concept describe the composition of complex dishes (such as `Lasagna`) by expressing them as a list of ⟨`BasicFood`, quantity⟩ pairs.

The root concept of the **activity** branch is the `PhysicalActivity` concept that contains 21 subclasses representing likewise categories and a total of 859 individuals each one referring to a different activity. For each activity, we provide the amount of calories consumed in 1 min for each kilogram of user's weight and the MET (metabolic equivalent of task) value expressing the energy cost of the activity.

The **monitoring** branch models the knowledge enabling the whole monitoring activity of users' behaviors. This branch contains two main root concepts: `MonitoringRule` and `Violation`. The `MonitoringRule` concept provides a structured representation of the parameters inserted by the domain experts for defining how users should behave, according to a fixed structure (aka "rule template"). Monitoring rules operate either on (1) a single user's meal or physical activity event, e.g., to check if they exceeded expert prescriptions (QB-Rules), (2) on user's events collected during a whole day (DAY-Rules), or (3) on user's events of a whole week (WEEK-Rules), to account for misbehaviors defined on a longer time scale.[2] `Violation` instances describe the results of the reasoning activities, and they can be exploited for generating users' advises and recommendations. The content of each `Violation` instance is computed according to the user data that triggered the violation.

The **user** branch contains the conceptualization of user information. This branch contains concepts enabling the representation of all users' events (food consumption and performed physical activities) and the linking of each violation to the corresponding user. Users' events are represented via the `Meal`, `ConsumedFood`, and `PerformedActivity` concepts. The last two concepts are reified relations

[2]The system supports the definition of customized timespans if necessary.

enriched with attributes for representing the facts that a user consumed a specific quantity of a food or performed an activity for a specific amount of time.

The ontology is publicly available including both TBox and ABox (with the exception of users' personal data, for privacy reasons). A RESTful interface is offered within HeLiS to query the ontology and ease its reuse within third-party applications.

5.2 Experts Support Facilities

The discussed platform integrates a set of facilities supporting domain experts in defining monitoring rules.

Here, it is necessary to clarify what we mean for *rule*. In logic, a *rule* (that in our case corresponds to a *semantic entailment*) is represented as a set of premises X that, if satisfied, lead to a conclusion Y: $X \models Y$. In our work, domain experts are in charge of modeling what can be called *domain rule*. By considering as example the Mediterranean diet, a domain rule is the quantity of vegetables that a person should eat every day. If we translate a domain rule into the logical representation shown above, it corresponds to the premises of the entailment. This means that in our architecture the experts provide only the premises of the entailment. Indeed, given the infinite combinations of data that can be provided by a user, the conclusion of the entailment (i.e., a violation) cannot be exactly defined a priori. For simplicity, hereafter with the term *rule*, we mean the premises of the entailment that are defined by the experts.

Rules are represented through rule templates, and domain experts have only to provide the parameters instantiating each rule template with the actual values. This way, domain experts do not need to learn the formal language for writing the monitoring rules. Here, we show the implemented facility supporting the conversion of the parameters given as input by the domain experts into a `MonitoringRule` instance.

5.3 Rule-Based Reasoning

Reasoning performed on the HeLiS ontology enriched with the data provided by users has the goal of verifying if user's lifestyle (i.e., eating behavior and physical activity) is consistent with the monitoring rules defined by domain experts, detecting, and possibly materializing violations in the knowledge base, upon which further actions may be taken. Reasoning is triggered each time a user's profile, associated meals, or performed activity reports are added or modified in the system and also at specific points in time (e.g., the end of a day or week), to check a user's behavior in such timespans. Although infrequent, changes to the monitoring rules, food, or nutrient parts of the ontology also trigger reasoning.

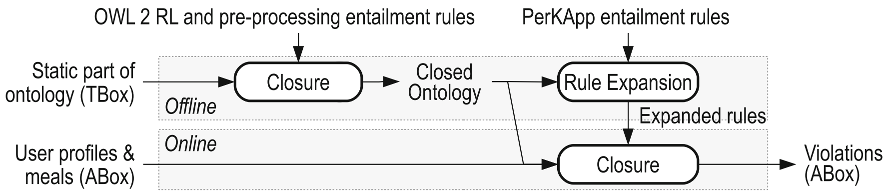

Fig. 3 Representation of the reasoning workflow

We implement reasoning using RDFpro,[3] a tool that allows us to provide out-of-the-box OWL 2 RL reasoning, supporting the fixed point evaluation of `INSERT...` `WHERE...` SPARQL-like entailment rules that leverage the full expressivity of SPARQL (e.g., `GROUP BY` aggregation, negation via `FILTER NOT EXISTS`, derivation of RDF nodes via `BIND`).

We organize reasoning in two *offline* and *online* phases as shown in Fig. 3. Offline, a one-time processing of the *static* part of the ontology (monitoring rules, food, nutrients, and activities) is performed to materialize its deductive closure, based on OWL 2 RL and some additional preprocessing rules that identify the most specific types of each `Nutrient` individual (this information greatly helps in aggregating their amounts).

Online, each time the reasoning is triggered (e.g., a new meal or performed activity is entered), the user data is merged with the closed ontology and the deductive closure of the *expanded rules* is computed. This process can be performed both on a per-user basis and globally on the whole knowledge base. The resulting `Violation` individuals and their RDF descriptions are then stored back in the knowledge base.

The online reasoning activity is in turn split in two further sessions: a *real-time reasoning* and a *background reasoning*. This is necessary due to the different kind of rules that the experts integrated into the platform. For example, by considering the Mediterranean diet, we have a total of 221 rules split in three different sets:

- QB-Rules: these rules define, for each food category contained in a rule, the right amount that should be consumed in a meal (if the food is consumed during a meal). These rules allow to monitor if a user exceeded the recommended amount of a specific food during a meal or not.
- DAY-Rules: these rules define, for each food category contained in a rule, the maximum (or minimum) amount (or number of portions) of the specified category that can be consumed during a single day. These rules allow to monitor the behavior of a user by aggregating foods consumed during an entire day.
- WEEK-Rules: these rules define, for each food category contained in a rule, the maximum (or minimum) amount (or number of portions) of the specified

[3] https://rdfpro.fbk.eu.

category that can be consumed during a week. These rules allow to monitor the behavior of a user by aggregating foods consumed during an entire week.

Similarly, concerning the physical activity domain, we integrated a set of QB-Rules defining the minimum amount of time which physical activities should last, a set of DAY-Rules defining the minimum amount of time that a user should dedicate to physical activities during a day, and finally a set of WEEK-Rules defining the minimum amount of time that a user should dedicate to physical activities during a week.

The time necessary for completing the reasoning over the different sets of rules is different based on the amount of data that has to be analyzed. Thus, in order to maintain the system efficient, we scheduled the reasoning activity according to the two sessions mentioned above. The *real-time reasoning* operates on the set of QB-Rules enabling the possibility of providing an immediate feedback to users based on the content of their last meal. This kind of reasoning suffers from the possibility of high concurrency due to the amount of people providing their data during a small time interval. Hence, by reducing as much as possible the number of rules evaluated by the reasoner, we are able to manage potential bottlenecks in elaborating data.

On the contrary, the *background reasoning* is performed on rules that have to be evaluated on aggregated sets of data in order to provide, eventually, violations about incorrect behaviors monitored during a medium or a long period of time. The *background reasoning* works on both the DAY-Rules and WEEK-Rules sets. The evaluation of these rules implies the collection and aggregation of a relevant amount of data requiring several time for being analyzed. The evaluation of these rule sets has to be scheduled for time slots with a small number of requests to avoid affecting the performance of the entire system.

The result of the reasoning activity is a set of structured packages, representing instances of the `Violation` concept. These packages contain specific information about the detected violations. Besides information directly inherited by the `MonitoringRule` instance associated with the violation for each violation, the package contains:

- the list of meals contributed to generate the violation. If the violated rule belongs to the QB-Rules set, the list will contain only one meal's reference, while if the violated rule belongs to either the DAY-Rules or to the WEEK-Rules sets, the list may contain more than one meal's reference;
- the actual quantity provided by the user;
- the expected quantity;
- the violation level. This value gives a dimension of the violation. The higher the gap between the actual and the expected values is, the higher the value of the violation level parameter will be;
- the violation history. The reasoner computes this value in order to provide a recidivism index about how a user is inclined to violate specific rules.

These information, together with the identifiers of rule and user, the rule priority, and the reference of the food (or food category, or nutrient) violated by the user, are sent to the *Dialog-Based Persuasive Layer* that elaborates these packages and decides which information to use for generating the feedback that has to be sent to the user.

6 PersEO: The Dialog-Based Persuasive Layer

The goal of PersEO (Persuasive mEssage generatOr) is the generation of dialogs for motivating users to adopt healthy lifestyles. This component is in charge of composing contextualized messages based on the users' data (both explicitly provided and implicitly acquired from sensors) and managing the dialog unfolding according to the responses provided by users to system utterances. This component is based on a state machine implemented in Drools,[4] the business rules management system (BRMS) solution with a forward and backward chaining inference-based rules engine. In this version of the platform, a dialog is represented as a directed acyclic graph (DAG), in which the vertexes are the single text messages sent by the system to the user (system utterance); see an example in Fig. 4. Each system utterance can be either a motivational message, which does not require an answer, or a question, possibly accompanied by a motivational part. In the former case, the utterance can be a leaf vertex and the dialog ends, till the next interaction triggered

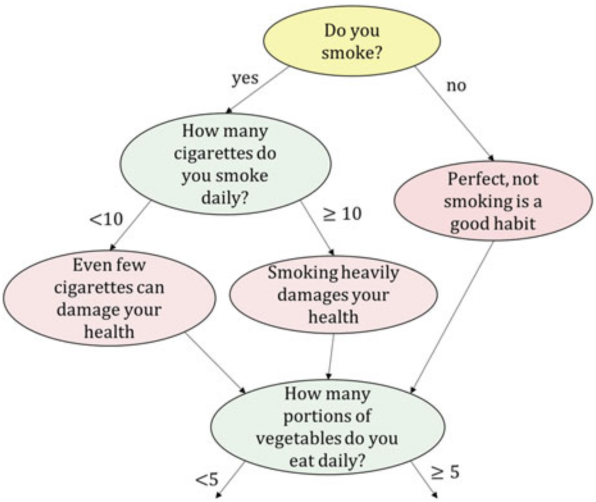

Fig. 4 A fragment of a DAG representing a dialog for profiling the user dialog regarding his/her lifestyle habits, with question messages that require a categorical answer and a numeric one (in yellow and green, respectively) and motivational messages (in red)

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

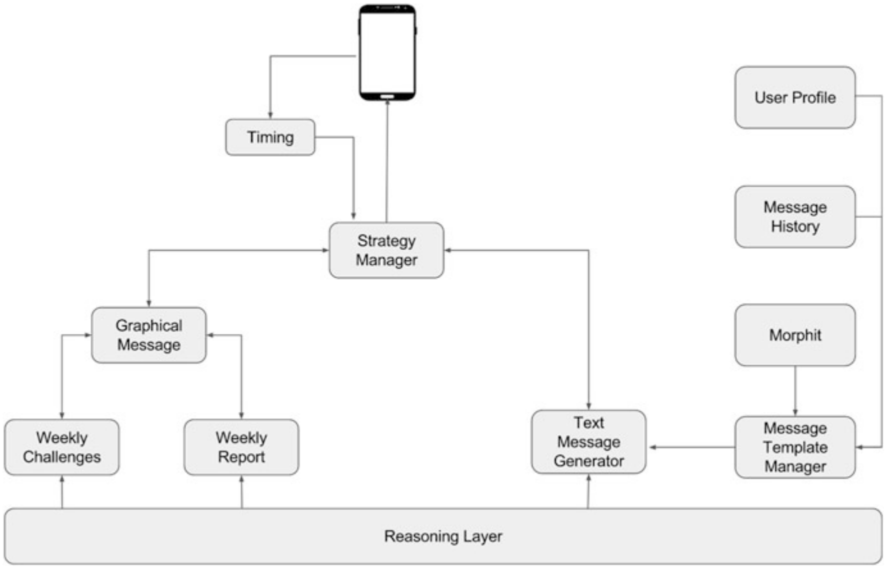

Fig. 5 The Persuasive Layer model

by PersEO or by the user, or there can be an edge to the next message or question of the same dialog. In the latter case, each possible answer to the question corresponds to an edge that connects the message to the next system utterance to send to the user. In this version, we modeled two kinds of answers to a question:

1. Closed-ended questions, which require the user to choose among a set of predefined questions (e.g., *Do you smoke? Yes/No*). These will be represented in the user interfaces by buttons or a list of possible choices.
2. Questions that require a numerical answer (e.g., *How many cigarettes do you smoke daily?*). The answer is elaborated by PersEO using the comparison operators ($\leq$, $\geq$, =, etc.) to pick up the next message according to the conditions formalized in the corresponding DAG vertex.

A motivational message (or the motivational part of a question) can be predefined or context-dependent composed at runtime by the motivational engine. In particular, a message can be generated according to the (1) timing of the message generation trigger, (2) level of violation of the violated rules, (3) information that the user can be interested in, and (4) history of previous messages sent to the user. The combination of these elements represents the *context* of the message. Below, we describe more in detail the persuasive model (Fig. 5) focusing on the four factors mentioned above and on the meta-reasoning implemented for each of them.

6.1 Timing

Timing represents the event prompting PersEO to create a new message. In our case study, message generation is triggered by specific events detected by the mobile application (Input Layer). Here we considered only system instantiated timing [17]; contextualization, tailoring, and efficacy of the message depend heavily on this aspect. For this reason, PersEO executes a meta-reasoning to evaluate if a message generation is needed and which form of message is more appropriate in that particular moment. There are three kinds of events detected by the Input Layer:

- Events related to user's habits and behavior: in general a behavior is analyzed when a user inputs data in the system, such as a new meal in the food diary (Fig. 6a).
- Time scheduling: PersEO may need to send particular information to the user at a specific time of the day or of the week (i.e., every Sunday at 18 p.m., the user receives a report about weekly adherence to the Mediterranean diet) or to perform a data input check to, eventually, send reminders to the user (e.g., if at 2 p.m. no lunch was added, PersEO invites the user to do it) (Fig. 6b). In this case, scheduling is defined observing user routine.
- Localization: the third event triggering the intervention of PersEO is the mobile application recognizing that the user is in a specific place (e.g., near a vending

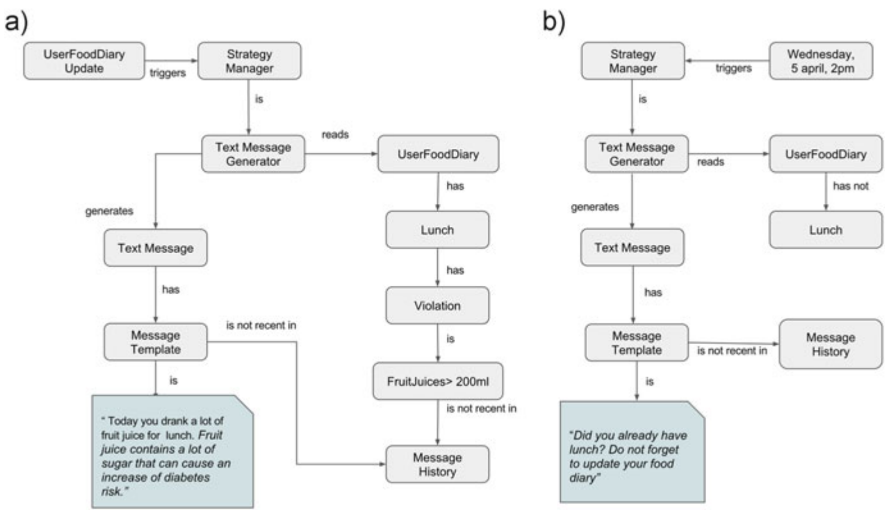

Fig. 6 Examples of message generation workflow. On the left the generation of a message triggered by a user-generated event (the recording of a meal). In this case the system controls the presence of violations and generates the message according to violation type and message history. On the right the message is triggered by a scheduled time. In this case, since the system does not find information, the message is a reminder to insert a meal. (**a**) Persuasion Engine generates a post feedback message. (**b**) Persuasion Engine generates a reminder message

machine). Even in this case, the generation of a message depends on the event time. For example, if the position in front of a vending machine is detected midmorning, it is highly probable that the user is going to have a snack.

Timing type determines the form and the structure of the message. In the first case, message is considered as a *post* strategy, while in the second and third, messages could be generated as a *pre* strategy.

6.2 Choice of Violation

Messages should provide feedback to the user about his/her eating and physical exercise behavior, according to modeled rules. Messages generated following the detection of violations are, in general, those with *negative* feedback. Following an event that triggered message generation, PersEO asks to the Knowledge Layer the list of violations generated. The violation bean contains the information needed to determine the behavior of a user. For example, a violation of a diet rule includes the entities that generated an unhealthy behavior (meal and food), the rule priority, and the number of times the same violation has been committed (history). If the list of violations is empty, the system can conclude that the user adopted a healthy behavior so it can decide to send messages with *positive* reinforcing feedback. If the list of violations is not empty, we decided to send a message regarding only one violation to avoid to annoy the user with repetitive information on one hand and provide messages with varied content informing the user about different aspects of correct behavior, on the other. The violation is chosen according to (1) its priority, (2) the number of times it was committed (recorded in the history parameter), and (3) the number of times the same violation was the object of a message. For example, if a message discouraging user to drink fruit juice has been already sent in the last 4 days, the persuasive engine decides to consider another violation with the same priority or the next highest present in the violation package but not sent recently. No message is generated if no eligible violation is detected.

6.3 Message Composition

Continuing with the diet example, after the choice of the violation, PersEO has the following information: (1) the user updated his/her food diary adding the list of foods eaten during lunch (timing), and (2) there are no messages sent in the recent past to the user that contained feedback about fruit juice (message history). Based on this information, the system decides the structure and the text content of the message.

The structure of the message, inspired by the work in [17] and expanded taking into consideration additional strategies presented in [24], consists of several

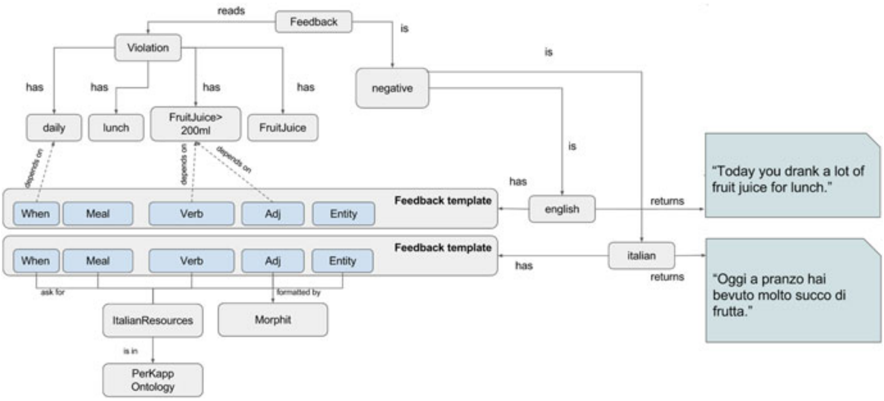

Fig. 7 Model for generating the text of feedback. The choices of template and message chunks depend on the violation. Different languages entail different linguistic resources. This holds also for both argument and suggestion

persuasion strategies that can be combined together to form a complex message. Here we will focus on three main parts: feedback, argument, and suggestion. Their generation follows the schema described in Sect. 4.3. For each part of the message, there is a template instantiating it according to the desired language.[5]

Below we describe the strategies implemented to automate the message generation, focusing also on linguistic choices:

Feedback is the part of the message that informs the user about his/her unhealthy behavior. Feedback is generated considering data included in the selected violation: entity of the violation will represent the object of the feedback, while the level of violation (e.g., deviation between food quantity expected and that actually taken by the user) is used to represent the severity of the incorrect behavior. Feedback contains also information about timing to inform the user about the moment in which violation was committed (Fig. 7). From a linguistic point of view, choices in the feedback are related to the verb and its tense: e.g., beverages imply use of the verb *to drink*, while for solid food we used *to eat*. To increase the variety of the message, the verbs *to consume* and *to intake* are also used. Simple past tense is used when violation is related to a specific moment (e.g., *You drank a lot of fruit juice for lunch*), while simple present continuous is used when the violation is related to a period of time of more days and the period has not yet ended (e.g., *You are drinking a lot of fruit juice this week*).

Argument is the part of the message that informs the user about the possible consequences of a behavior. For example, in the case of diet recommendations, argument consists of two parts: (1) information about nutrients contained in the food intake that caused the violation and (2) information about consequences that

[5]The current version of PersEO supports the generation of messages in English and Italian.

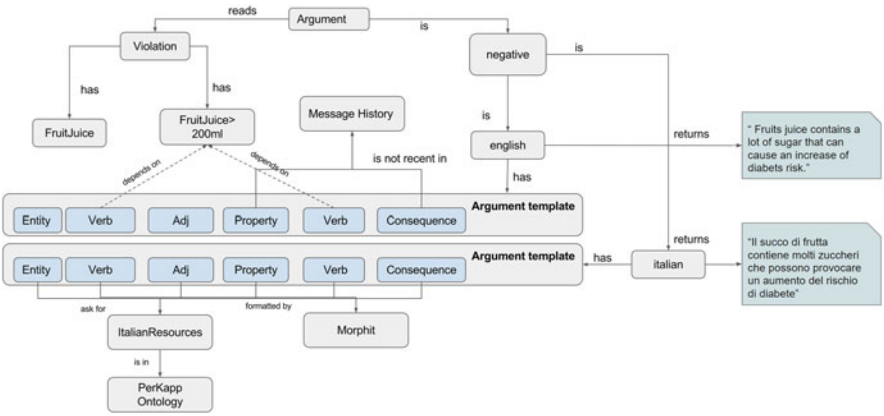

Fig. 8 Model for generating text of argument

nutrients have on human body and health. Consequences imply the positive or negative aspects of nutrients. In this case, PersEO uses the rule constraint contained in the selected violation to identify the type of argument to generate. Considering the example of violation above, constraint *less* (fruitjuice $<=$ 200 ml) implies that an excess of this food can cause negative consequences on user health, due to an excess of a particular nutrient of this food. Hence, the system needs to ask for *negative* nutrients and *negative* consequences to the Knowledge Layer. On the contrary, constraint *greater* (vegetables $>=$ 200 g) implies that the body has many advantages from getting nutrients contained in that food; so *positive* nutrients and *positive* consequences are asked to the Knowledge Layer.

Moreover, PersEO analyzes the message history to decide if a property returned by the Knowledge Layer in the violation bean can be used in the argument. Similar to the approach followed in choosing a violation, properties are eligible for argument text only if they were not in the text of a message sent in the past few days. With respect to the linguistic choices, the type of nutrients and their consequences influence the verb usage in the text. To emphasize negative aspects of the food, we used the verb *contain* for nutrients and *can cause* for the consequences. Positive aspects are highlighted by the phrase *is rich in* and the verb *help* used for nutrients and consequences, respectively (Fig. 8).

Suggestion This part represents the solution that PersEO wants to deliver to users in order to motivate them to change their behavior. The model for generating a suggestion message is shown in Fig. 9. Exploiting the information available, described at the beginning of this section, PersEO generates a *post* suggestion to inform the user about the alternative and healthy behavior that he/she can adopt. To do that, the data contained in the selected violation are not sufficient. PersEO performs an additional meta-reasoning to identify the appropriate content that depends on (1) qualitative properties of food, (2) user profile, (3) other specific violations, and (4) history of messages sent.

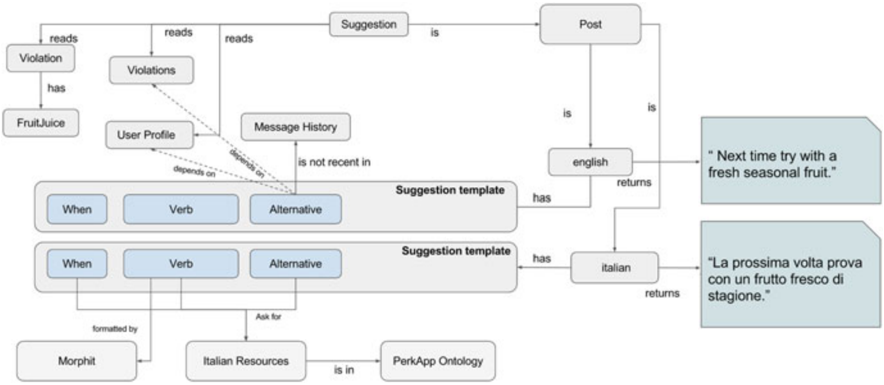

Fig. 9 Model for generating text of suggestion

First of all, the system asks the Knowledge Layer to provide a list of foods having properties that render them valid alternatives to the consumed food (e.g., similar-taste relation, list of nutrients, consequences on user health). These alternatives are firstly filtered according to the user profile: the system will exclude all the foods that cannot be consumed by people belonging to certain profiles. Considering the vegetarians, for example, the system cannot invite this category of people to consume fish as an alternative to legumes, even if the former is an alternative to the latter when one considers only the nutrients. An additional filter is applied on alternative foods. The system cannot suggest the consumption of foods that can cause a violation of the type *less* or *equal*, because this can generate a contradiction with healthy behavior rules. For example, the system cannot recommend meat as alternative to cheese as a source of animal proteins, when a rule sets a maximum quantity of meal.In general this control has more sense when pre-suggestion are created. Finally, control on messages history is again executed, with the same rules described above. Regarding the linguistic aspect, the system uses the verbs *try* and *alternate* to emphasize the alternative behavior.

7 Platform Validation

The validation and evaluation of our platform have been tested through a user study designed within Fondazione Bruno Kessler. The user study consisted in providing to a group of users a mobile application we created based on the services included into our platform. We analyzed the usage of a mobile application connected with our platform for 7 weeks by monitoring information provided by the users and the associated violations, if any. Our goal was to measure the effectiveness of the persuasive messages generated by our platform by observing the evolution of the number of detected violations. This analysis has been performed by considering the data provided by the 92 users participating in the user study all selected among the employees of the Fondazione Bruno Kessler. In order to validate the effectiveness of the persuasive messages, we also run a control group composed of further 27 users that used the same mobile application for the same timespan. Users of the control group did not receive feedback generated by PersEO but only canned text messages notifying if a rule has been violated. The expectation was to find a higher decrease in the number of violations through time by the users receiving persuasive messages.

All users have reported their meals on a regular basis (i.e., five times a day for a period of 49 days), while their physical activities have been reported only occasionally. For this reason, we focus our violation analysis only on the meal data. The fact that physical activity data have been reported only occasionally was not associated with a low usability aspect of the mobile application but the availability of personal pedometer bracelets. Actually, those who had one of such devices provided data on a regular basis, but their number was too low to allow for a significant analysis (even if the trend on the number of detected violations in physical activity is consistent with the dietary one). It will be part of the future work to improve data collection about physical activities.

Table 1 shows main demographic information concerning the users involved in the performed evaluation campaign. All users presented a healthy status. Indeed, in this first pilot, we decided to do not involve people affected by chronic diseases or other pathologies.

Results concerning the evolution of the violation numbers are presented in Fig. 10. The three graphs show the average number of violations per user related to the QB-Rules, DAY-Rules, and WEEK-Rules sets, respectively. Blue line represents the number of violations while the red line the average standard deviation observed for each single event. Then, the green line represents the average number of violations generated by the control group and the orange line the associated standard deviation. As mentioned earlier, QB-Rules are verified every time a user stores a meal within the platform; DAY-Rules are verified at the end of the day, while WEEK-Rules are verified at the end of each week. The increasing trend of the gap between the blue and green lines demonstrates the positive impact of the persuasive

Table 1 Distribution of demographic information of the users involved in the evaluation campaign

Dimension	Property	Value
Gender	Male	57%
	Female	43%
Age	25–35	12%
	36–45	58%
	46–55	30%
Education	Master’s degree	42%
	Ph.D. degree	58%
Occupation	Ph.D. student	8%
	Administration	28%
	Researcher	64%

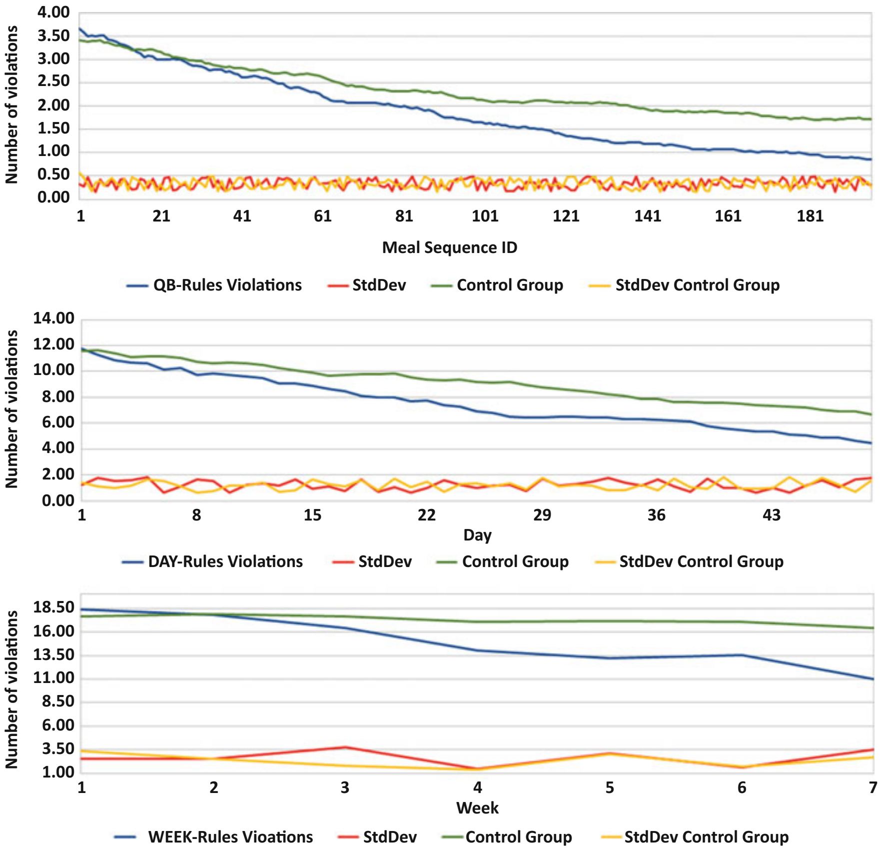

Fig. 10 Evolution of the number of detected violations through the Key To Health project timespan

messages sent to users. We can observe how for the QB-Rules the average number of violations is below 1.0 after the first 7 weeks of the project. This means that some users started to follow all the guidelines about what to consume during a single meal. A positive result has been obtained also for the DAY-Rules and the WEEK-Rules. In particular, for what concerns DAY-Rules, the average number of violations per user at the end of the observed period is acceptable by considering that it drops of about 67%. For the WEEK-Rules, however, the drop remained limited. By combining the evolution of the number of violations with the demographic information shown in Table 1, we did not find any particular correlation worthy of discussion. By considering the standard deviation lines, we can appreciate how both lines remain contained within low bounds. Indeed after a more in-depth analysis of the data, we did not observe the presence of outliers.

In order to deeply analyze this fact, we organized a focus group with the users. During the discussion, we discovered that several users perceived the combination

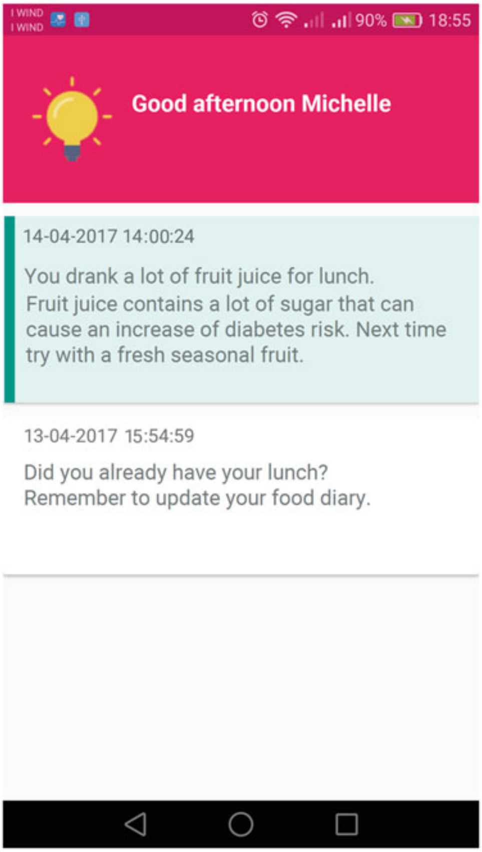

Fig. 11 A screenshot of the interface with the history of all messages received by the user. The highlighted one is the most recent message received, and it is a "post" message on an unhealthy behavior

of some rules very hard to follow. Examples of such rules were the ones related to *vegetables* (at least three times a day) and the consumption of *milk and yogurt* (at least once a day). In the first case, many users found hard to introduce the third portion of vegetables within their daily diet. In the second case, some users experienced a psychological barrier concerning the consumption of such a food category due to their fear of having some digestion problems. We reported these feedbacks to the experts that took them into account for a new refinement iteration of the monitoring rules that will be implemented in the future deployments of the platform.

Figures 11 and 12 show a couple of screenshot of the mobile application available to users. In particular here we show two examples of the textual interaction between the users and the mobile application.

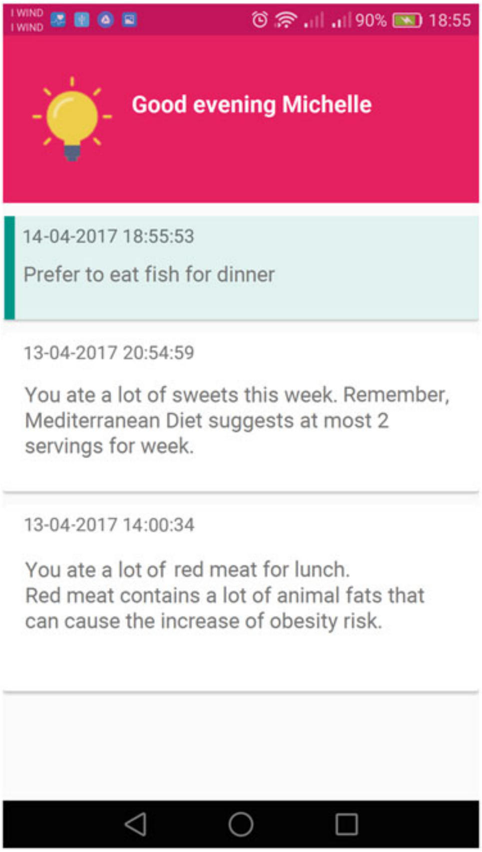

Fig. 12 A screenshot of the interface with the history of all messages received by the user. The highlighted one is the most recent message received, and it is a "suggestion" message on a desirable behavior

In addition to text-based realization, other representations have been integrated. Graphical elements and charts are used to represent user adherence to a healthy behavior. In particular, we used an HGraph-based representation[6] (see Fig. 13) and a score chart (see Fig. 14) to inform user about his weekly adherence to the Mediterranean diet. Score is calculated considering all the violations committed by the user during the week and their violation level.

Finally, we show in Table 2 examples of questions that have been submitted to users after the pilot with some of the collected answers.

[6] www.hgraph.org.

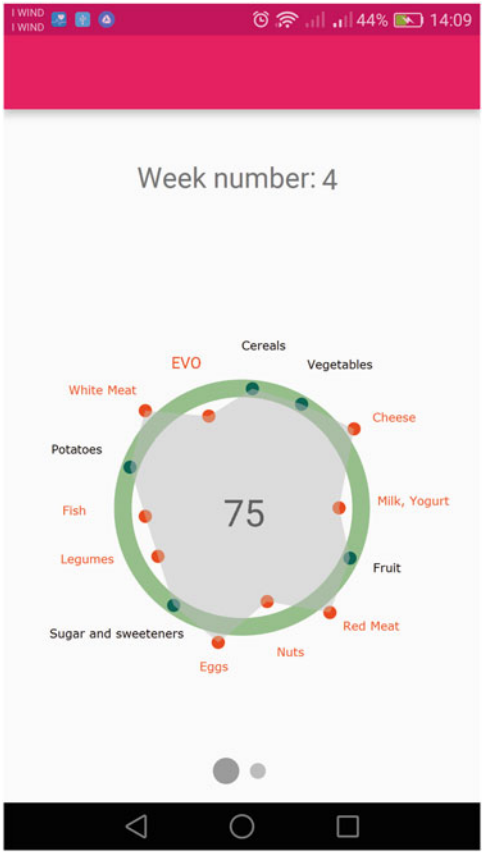

Fig. 13 The HGraph in the screenshot represents the Mediterranean diet adherence of the user over the correspondent week

8 Conclusions

The contribution presented in this chapter focused on the design and implementation of a persuasive platform able to monitor people's habits from both the dietary and physical activity perspectives. The platform had to motivate them to change their behaviors through the interaction supported by messages automatically generated, with the final goal of persuading them into following healthier lifestyles.

We presented and discussed the challenges that need to be addressed from both the psychological and technological perspectives in order to build an effective persuasive tool. In particular, we presented the overall architecture by describing the main technological blocks and how they are connected.

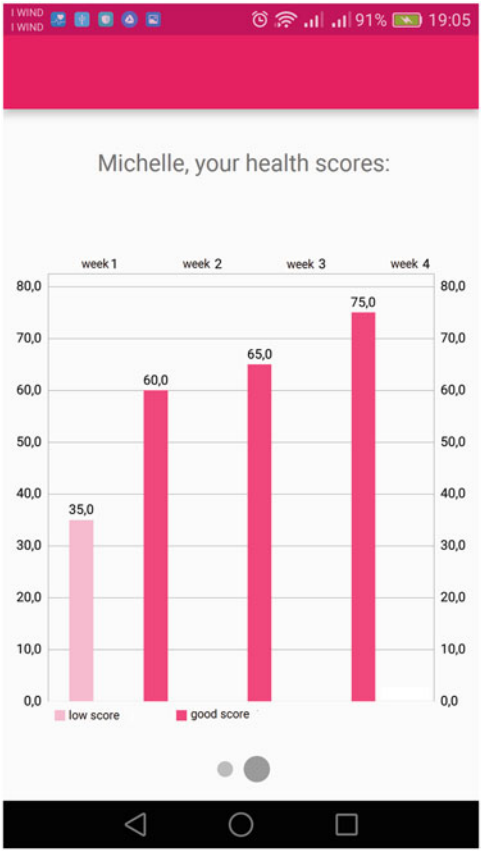

Fig. 14 A screenshot of the history of weekly scores achieved by the user. Scores represent the Mediterranean diet adherence of the user over time

We described how the use of knowledge bases has been integrated in order to provide a structured and precise representation of heterogeneous information for supporting the generation of persuasive messages. Then, we presented how the generation of persuasive conversations benefits from the output provided by the use of knowledge bases and how these persuasive technologies have been deployed by detailing the pipeline implemented for supporting the generation of the persuasive messages delivered to users.

Finally, we introduced how the work presented in this chapter and the experience collected from a pilot user study leave room to future enrichments of our platform.

Table 2 Examples of questions submitted to users after the pilot with the related corrected answers

Question	Answers
How did your daily routine change during the last 7 weeks?	I increased the consumption of vegetables
	I discovered the importance of making a rich breakfast
	I appreciate the lightness of eating fish
Which aspect of the mobile application encouraged you to continue the behavior change path?	The content of the messages was varying
	The HGraph is very useful for understanding my adherence to the diet
Which facilities would you change in the mobile application?	To include more education information into the provided messages
	To add graphs providing information about single nutrients
Which were the difficulties encountered during the usage of the mobile application?	In some cases the application did not provide the feedback in reasonable time
	Some recipes are not included in the application

References

1. Elwood, P., Galante, J., Pickering, J., Palmer, S., Bayer, A., Ben-Shlomo, Y., Longley, M., Gallacher, J.: Healthy lifestyles reduce the incidence of chronic diseases and dementia: evidence from the Caerphilly cohort study. PLoS ONE **8**(12), e81877 (2013)
2. Booth, F.W., Roberts, C.K., Laye, M.J.: Lack of exercise is a major cause of chronic diseases. Compr. Physiol. **2**(2), 1143–1211 (2012)
3. Gualtieri, L., Rosenbluth, S., Phillips, J.: Can a free wearable activity tracker change behavior? The impact of trackers on adults in a physician-led wellness group. JMIR. Res. Protoc. **5**(4), e237 (2016)
4. Fogg, B.J.: Persuasive Technology: Using Computers to Change What We Think and Do. Morgan Kaufmann Publishers, Burlington (2002)
5. Maimone, R., Guerini, M., Dragoni, M., Bailoni, T., Eccher, C.: Perkapp: a general purpose persuasion architecture for healthy lifestyles. J. Biomed. Inform. **82**, 70–87 (2018)
6. Dragoni, M., Bailoni, T., Eccher, C., Guerini, M., Maimone, R.: A semantic-enabled platform for supporting healthy lifestyles. In: Seffah, A., Penzenstadler, B., Alves, C., Peng, X. (eds.) Proceedings of the Symposium on Applied Computing, SAC 2017, Marrakech, Morocco, 3–7 April 2017, pp. 315–322. ACM, New York (2017)
7. Bailoni, T., Dragoni, M., Eccher, C., Guerini, M., Maimone, R.: Perkapp: a context aware motivational system for healthier lifestyles. In: IEEE International Smart Cities Conference, ISC2 2016, Trento, 12–15 Sept 2016, pp. 1–4. IEEE, Piscataway (2016)
8. Dragoni, M., Rospocher, M., Bailoni, T., Maimone, R., Eccher, C.: Semantic technologies for healthy lifestyle monitoring. In: Gonçalves, R., Kaffee, L.A. (eds.) Proceedings of 17th International Semantic Web Conference, ISWC 2018, Monterey, 8–12 Oct 2018
9. Oinas-Kukkonen, H.: A foundation for the study of behavior change support systems. Pers. Ubiquit. Comput. **17**(6), 1223–1235 (2013)
10. Kelders, S., Oinas-Kukkonen, H., Oörni, A., van Gemert, J.E.W.C.: Health behavior change support systems as a research discipline; a viewpoint. Int. J. Med. Inform. **96**, 3–10 (2016)

11. Kaptein, M., de Ruyter, B.E.R., Markopoulos, P., Aarts, E.H.L.: Adaptive persuasive systems: a study of tailored persuasive text messages to reduce snacking. ACM Trans. Interact. Intell. Syst. **2**(2), 10:1–10:25 (2012)
12. Oinas-Kukkonen, H., Harjumaa, M.: Persuasive systems design: key issues, process model, and system features. Commun. Assoc. Inf. Syst. **24**, 28 (2009)
13. Zukerman, I., McConachy, R., Korb, K.: Using argumentation strategies in automated argument generation. In: Proceedings of the 1st International Natural Language Generation Conference, pp. 55–62 (2000)
14. Reed, C., Long, D., Fox, M.: An architecture for argumentative dialogue planning. In: Practical Reasoning: Proceedings of the First International Conference on Formal and Applied Practical Reasoning (FAPR96), pp. 555–566. Springer, Berlin (1996)
15. Pan, S., Zhou, M.: Pplum: a framework for large-scale personal persuasion. In: Proceedings of the 3rd Workshop on Data-Driven User Behavioral Modeling and Mining from Social Media, pp. 5–6. ACM, New York (2014)
16. Guerini, M., Stock, O., Zancanaro, M., O'Keefe, D.J., Mazzotta, I., de Rosis, F., Poggi, I., Lim, M.Y., Aylett, R.: Approaches to verbal persuasion in intelligent user interfaces. In: Emotion-Oriented Systems, pp. 559–584. Springer, Berlin (2011)
17. op den Akker, H., Cabrita, M., op den Akker, R., Jones, V.M., Hermens, H.: Tailored motivational message generation: a model and practical framework for real-time physical activity coaching. J. Biomed. Inform. **55**, 104–115 (2015)
18. Berrouiguet, S., Baca-García, E., Brandt, S., Walter, M., Courtet, P.: Fundamentals for future mobile-health (mHealth): a systematic review of mobile phone and web-based text messaging in mental health. J. Med. Internet Res. **18**(6), e135 (2016)
19. Head, K., Noar, S., Iannarino, N., Harrington, N.: Efficacy of text messaging-based interventions for health promotion: a meta-analysis. Soc. Sci. Med. **97**, 41–48 (2013)
20. Achterkamp, R., Weering, M.D., Evering, R.M., Tabak, M., Timmerman, J.G., Hermens, H.J., Vollenbroek-Hutten, M.M.R.: Strategies to improve effectiveness of physical activity coaching systems: development of personas for providing tailored feedback. Health Inform. J. **24**(1), 92–102 (2018)
21. Job, R.J., Spark, C.L., Fjeldsoe, S.B., Eakin, G.E., Reeves, M.M.: Women's perceptions of participation in an extended contact text message–based weight loss intervention: an explorative study. JMIR Mhealth Uhealth **5**, e21 (2017)
22. Wabeke, T.R.: Recommending tips that support wellbeing at work to knowledge workers. Master thesis. https://theses.ubn.ru.nl/bitstream/handle/123456789/224/Wabeke%2C_T.R._1.pdf?sequence=1
23. Anselma, L., Mazzei, A., De Michieli, F.: An artificial intelligence framework for compensating transgressions and its application to diet management. J. Biomed. Inform. **68**, 58–70 (2017)
24. Guerini, M., Stock, O., Zancanaro, M.: A taxonomy of strategies for multimodal persuasive message generation. Appl. Artif. Intell. J. **21**(2), 99–136 (2007)
25. Bailoni, T., Dragoni, M., Eccher, C., Guerini, M., Maimone, R.: Healthy lifestyle support: the perkapp ontology. In: Dragoni, M., Poveda-Villalón, M., Jiménez-Ruiz, E. (eds.) OWL: - Experiences and Directions - Reasoner Evaluation - 13th International Workshop, OWLED 2016, and 5th International Workshop, ORE 2016, Bologna, 20 Nov 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 10161, pp. 15–23. Springer, Berlin (2016)
26. Dragoni, M., Bailoni, T., Maimone, R., Eccher, C.: Helis: an ontology for supporting healthy lifestyles. In: Gonçalves, R., Kaffee, L.A. (eds.) Proceedings of 17th International Semantic Web Conference, ISWC 2018, Monterey, CA, 8–12 Oct 2018