# Towards diet management with automatic reasoning and persuasive natural language generation

Luca Anselma and Alessandro Mazzei

Dipartimento di Informatica, Università di Torino, Italy
{anselma,mazzei}@di.unito.it

**Abstract.** We devise a scenario where the interaction between man and food is mediated by an intelligent system that, on the basis of various factors, encourages or discourages the user to eat a specific dish. The main factors that the system need to account for are (1) the diet that the user intends to follow, (2) the food that s/he has eaten in the last days, and (3) the nutritional values of the dishes and their specific recipes. Automatic reasoning and Natural Language Generation (NLG) play a fundamental role in this project: the compatibility of a food with a diet is formalized as a Simple Temporal Problem (STP), while the NLG tries to motivate the user. In this paper we describe these two facilities and their interface.

**Keywords:** diet management, automatic reasoning, natural language generation

## 1 Introduction

The daily diet is one of the most important factors influencing diseases, in particular for obesity. As highlighted by the World Health Organization, this factor is primarily due to the recent changes in the lifestyle [26]. The necessity to encourage the world's population toward a healthy diet has been sponsored by the FAO [20]. In addition, many states specialized these guidelines by adopting strategies related to their *food history* (for instance, for USA `http://www.choosemyplate.gov`). In Italy, the Italian Society for Human Nutrition has recently produced a prototypical study with recommendations for the use of specialized operators [1].

This scenario suggests the possibility to integrate the directives on nutrition in the daily diet of people by using multimedia tools on mobile devices. The smartphone can be considered as an super-sense that creates new modalities of interaction with food. In recent years there has been a growing interest in using multimedia applications on mobile devices as *persuasive* technologies [13].

Often a user is not able to carefully follow a diet for a number of reasons. When a deviation occurs, it is useful to support the user in devising the consequences of such deviation and to dynamically adapt the rest of the diet in the

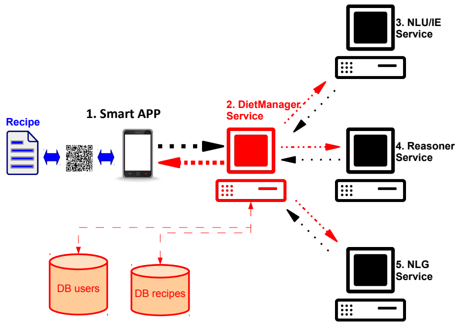


**Fig. 1.** The architecture of the diet management system.

upcoming meals so that the global Dietary Reference Values (henceforth DRVs) could nevertheless be reached. In particular in this paper we describe a system which is useful for (i) evaluating the compatibility of a dish with a diet allowing small and occasional episodes of diet disobedience, (ii) determining what are the consequences of eating a specific dish on the rest of the diet, (iii) showing such consequences to the user thus empowering her/him and, moreover, (iv) motivating the user in following the diet by persuading her/him to minimize the acts of disobedience. Using automatic reasoning to evaluate the compatibility of a dish with a diet could enhance a smartphone application with a sort of *virtual dietitian*. Artificial intelligence should make the system *tolerant* to diet disobedience, but also *persuasive* to minimize these acts of disobedience. Thus, a critical issue directly related to automatic reasoning is the final presentation to the user of the results. Several studies have addressed the problem of generating natural language sentences that explain the results of automatic reasoning [4, 17].

In our hypothetical scenario the interaction between man and food is mediated by an intelligent system that, on the basis of various factors, encourages or discourages the user to eat a specific dish. The main factors that the system needs to account for are (1) the diet that the user has to follow, (2) the food that s/he has been eating in the last days or that s/he intends to eat in the next days, and (3) the nutritional values of the ingredients of the dish and its specific recipe. In Fig. 1 we report the architecture of our system. It is composed by five modules/services: a smartphone application (APP), a central module that manages the information flow (DietMAnager), an information extraction module (NLU/IE), a reasoning module (Reasoner) and a natural language generation module (NLGenerator). In this paper we focus on the description of the Reasoner and NLGenerator modules; some details on the other modules and on the system can be found on the webpage of the project (`http://di.unito.it/madiman`).

We think that this system could be commercially attractive at least in two contexts. The first context is the medical one, where users (e.g. patients affected by essential obesity) are strongly motivated to strictly follow a diet and need tools that help them. The second context is the one involving, e.g., healthy fast food or restaurant chains, where the effort of deploying the system can be rewarded by an increase in customer retention.

This paper is organized as follows: in Section 2 we describe the automatic reasoning facilities, in Section 3 we describe the design of the persuasive NLG based on different theories of persuasion and, finally, in Section 4 we draw some conclusions.

## 2 Automatic Reasoning for Diet Management

Since our approach to automatic reasoning for diet management is based on the STP framework, first we introduce STP, then we describe how we exploit STP to reason on a diet and how we interpret the results from STP.

### 2.1 Preliminaries: STP

We base our treatment of nutrition constraints on the framework of "Simple Temporal Problem" (STP) [8]. An STP constraint consists in a bound on differences of the form $c \leq x - y \leq d$, where $x$ and $y$ are temporal points and $c$ and $d$ are numbers (their domain can be either discrete or real). An STP constraint can be interpreted in the following way: the temporal distance between the time points $x$ and $y$ is between $c$ – the lower bound of the distance – and $d$ – the upper bound of the distance. It is also possible to impose strict inequalities (i.e., $<$) and $-\infty$ and $+\infty$ can be used to denote the fact that there is no lower or upper bound, respectively. An STP is a conjunction of STP constraints.

An interesting feature of STP is that the problem of determining the consistency of an STP is tractable and that the algorithm employed, i.e., an all-pairs shortest paths algorithm such as Floyd-Warshall's one, also obtains the minimal network, that is the minimum and maximum distance between each pair of points. STP can be represented with a graph whose nodes correspond to the temporal points of the STP and whose arcs are labeled with the temporal distance between the points.

**Property.** Floyd-Warshall's algorithm is correct and complete on STP, i.e. it performs all and only the correct inferences while propagating the STP constraints [8], and obtains a minimal network. Its temporal computational cost is cubic in the number of time points.

### 2.2 Towards automatically reasoning on a diet

**Reasoning on DRVs.** In a diet it is necessary to consider parameters such as the total energy requirements and the specific required amount of nutrients and macronutrients such as proteins, carbohydrates and lipids. In particular in the literature it is possible to find systems of DRVs that are recommended to be followed for significant amounts of time. In the running example, without loss of generality we refer to the Italian values [1]. Such values have to be customized for the specific patients according to their characteristics. In particular, from

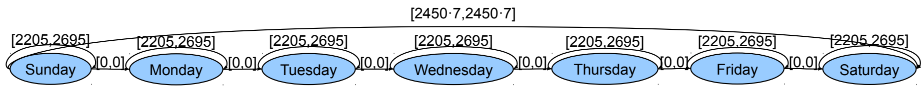


**Fig. 2.** Example of DRVs for a week represented as STP (for space constraints the constraints for the meals are not represented).

weight, gender and age, using Schofield equation [24], it is possible to estimate the basal metabolic rate; for example a 40-year-old male who is 1.80 m tall and weighs 71.3 kg has an estimated basal metabolic rate of 1690 kcal/day. Such value is then adjusted [1] by taking into account the energy expenditure related to the physical activity of the individual; for example a sedentary lifestyle corresponds to a physical activity level of 1.45, thus, in the example, since the physical activity level is a multiplicative factor, the person has a total energy requirement of 2450 kcal/day. Moreover, it is recommended [1] that such energy is provided by the appropriate amount of the different macronutrients, e.g., 260 kcal/day of proteins, 735 kcal/day of lipids and 1455 kcal/day of carbohydrates. In this section we focus on the total energy requirement; the macronutrients can be dealt with separately in the same way.

We represent the DRVs as STPs; more precisely, we use an STP constraint to represent – instead of temporal distance between temporal points – the admissible DRVs. Thus, e.g., a recommendation to eat a lunch of minimum 500 kcal and maximum 600 kcal is represented by the STP constraint $500 \leq lunch_E - lunch_S \leq 600$, where $lunch_E$ and $lunch_S$ represent the end and the start of the lunch, respectively.

Furthermore, we exploit the STP framework to allow a user to make small deviations with regard to the "ideal" diet and to know in advance what are the consequences of such deviations on the rest of the diet. Thus, we impose less strict constraints over the shortest periods (i.e., days or meals) and stricter constraints over the longest periods (i.e., months, weeks). For example the recommended energy requirement of 2450 kcal/day, considered over a week, results in a constraint such as $2450 \cdot 7 \leq week_E - week_S \leq 2450 \cdot 7$ and for the single days we allow the user to set, e.g., a deviation of 10%, thus resulting in the constraints $2450 - 10\% \leq Sunday_E - Sunday_S \leq 2450 + 10\%$, ..., $2450 - 10\% \leq Saturday_E - Saturday_S \leq 2450 + 10\%$ (see Fig. 2). For single meals we can further relax the constraints: for example the user can decide to split the energy assumption for the day among the meals (e.g., 20% for breakfast and 40% for lunch and dinner) and to further relax the constraints (e.g., of 30%), thus resulting in a constraint, e.g., $2450 \cdot 20\% - 30\% \leq Sunday\_breakfast_E - Sunday\_breakfast_S \leq 2450 \cdot 20\% + 30\%$.

**Representing and reasoning on the diet and the food.** Along these lines, it is possible to represent the dietary recommendations for a specific user. However, we wish to support such a user into taking advantage of the information regarding the actual meals s/he consumes. In this way, the user can learn what

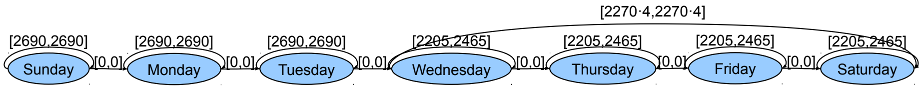


**Fig. 3.** Example of DRVs represented as STP.

are the consequences on his/her diet of eating a specific dish and s/he could use such information in order to make informed decisions about the current or future meals. Therefore it is necessary to "integrate" the information about the eaten dishes with the dietary recommendations. We devise a system where the user inputs the data about the food s/he is eating using a mobile app where the input is possibly supported by reading a QR code and s/he can also specify the amount of food s/he has eaten. Thus, we allow some imprecision due to possible differences in the portions (in fact, the actual amount of food in a portion is not always the same and, furthermore, a user may not eat a whole portion) or in the composition of the dish [6]. We support such feature by using STP constraints also for representing the nutritional values of the eaten food.

The dietary recommendations can be considered constraints on *classes*, which can be instantiated several times when the user assumes his/her meals. Thus, the problem of checking whether a meal satisfies the constraints of the dietary recommendations corresponds to checking whether the constraints of the instances satisfy the constraints of the classes. This problem has been dealt with in [25] and [2]. In these works the authors have considered the problem of "inheriting" the temporal constraints from classes of events to instances of events in the context of the STP framework, also taking into account problems deriving from correlation between events and from observability. In our setting we have a simpler setting, where correlation is known and observability is complete (even if possibly imprecise). Thus, we generate a new, provisional, STP where we add the new STP constraints deriving from the meals that the user has consumed: the added constraints possibly restrict the values allowed by the constraints in the STP. Then we propagate the constraints in such a new STP and we determine whether the new constraints are consistent and we obtain the new minimal network with the implied relations. For example, let us suppose that the user on Sunday, Monday and Tuesday had an actual intake of 2690 kcal for each day. This corresponds to adding to the STP the new constraints $2690 \leq Sunday_E - Sunday_S \leq 2690$, $\ldots$, $2690 \leq Tuesday_E - Tuesday_S \leq 2690$. Then, propagating the constraints of the new STP (see Fig. 3), we discover that (i) the STP is consistent and thus the intake is compatible with the diet and (ii) on each remaining day of the week the user has to assume a minimum of 2205 kcal and a maximum of 2465 kcal.

### 2.3 Interpretation of STP

Although the information deriving from the STP is complete (and correct), in order to show to the user a meaningful feedback and to make it possible to interface the automatic reasoning module with the NLG module, it is useful to interpret the results of the STP. In particular we wish to provide the user with a user-friendly information not limited to a harsh "consistent/inconsistent" answer regarding the adequacy of a dish with regard to her/his diet. Therefore we consider the case where the user proposes to our system a dish, we obtain its nutritional values, we translate them, along with the user's diet and past meals, into STP and, by propagating the constraints, we obtain the minimal network. By taking into account a single macronutrient (carbohydrates, lipids or proteins), the resulting STP allows us to classify the macronutrient in the proposed dish in one of the following five cases: *permanently inconsistent* (I.1), *occasionally inconsistent* (I.2), *consistent and not balanced* (C.1), *consistent and well-balanced* (C.2) and *consistent and perfectly balanced* (C.3).

In the cases I.1 and I.2 the value of the macronutrient is inconsistent. In case I.1 the value for the nutrient is inconsistent with the DRVs as represented in the user's diet. The dish cannot be accepted even independently of the other food s/he may possibly eat. This case is detected by considering whether the macronutrient violates a constraint on classes. In case I.2 the dish per se does not violate the DRVs, but – considering the past meals s/he has eaten – it would preclude to be consistent with the diet. Thus, it is inconsistent now, but it could become possible to choose it in the future, e.g., next week or month. This case is detected by determining whether the macronutrient, despite it satisfies the constraints on the classes, is inconsistent with the propagated inherited STP.

In the cases C.1, C.2 and C.3 the value of the macronutrient is consistent with the diet, also taking into account the other dishes that the user has already eaten. It is possible to detect that the dish is consistent by exploiting the minimal network of the STP: if the value of the macronutrient is included between the lower and upper bounds of the relative constraint, then we are guaranteed that the STP is consistent and that the dish is consistent with the diet. This can be proven by using the property that in a minimal network every tuple in a constraint can be extended to a solution [19]. A consistent but not balanced choice of a dish will have consequences on the rest of the user's diet because the user will have to "recover" from it. Thus we distinguish three cases depending on the level of the adequacy of the value of the macronutrient to the diet. In order to discriminate between the cases C.1, C.2 and C.3, we consider how the value of the macronutrient stacks upon the allowed range represented in the related STP constraint. We assume that the mean value is the "ideal" value according to the DRVs and we consider two parametric user-adjustable thresholds relative to the mean: according to the deviation with respect to the mean we classify the macronutrient as not balanced (C.1), well balanced (C.2) or perfectly balanced (C.3) (see Fig. 4). In particular, we distinguish between lack or excess of a specific macronutrient for a dish: if a macronutrient is lacking (in excess) with regard to the ideal value, we tag the dish with the keyword *IPO* (*IPER*). This information will be exploited in the generation of the messages.

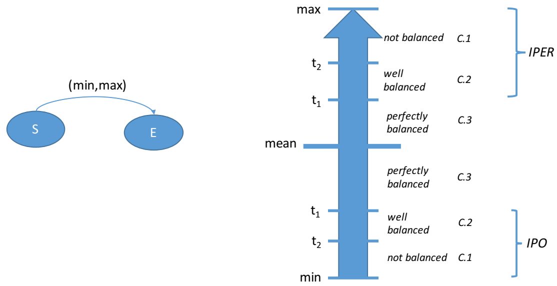


**Fig. 4.** Classification of a consistent value of a macronutrient given the minimum and maximum value of an STP constraint in a minimal network.

## 3 Persuasive NLG for diet

A number of works considered the problem of NLG for presenting the results of automated reasoning to a user, especially in the case of expert systems for reasoning, e.g., [4, 17]. In order to convert the five possible kinds of output of the STP reasoner (see Section 2.3) in messages, we adopted a simple template-based generator that produces five kinds of messages designed for persuasion. We first describe the generator (Section 3.1) and later we describe the theories that motivated the design of our messages (Section 3.2).

### 3.1 A simple template-based generation architecture

The standard architecture for NLG models generation is a pipeline composed by three distinct modules/processes: the document planning, the micro-planning, and the surface realization [22]. Each one of these modules addresses distinct issues, in particular: (1) In the document planning one decides what to say, that is which information contents will be communicated; (2) In the micro-planning, the focus is on the design of a number of features that are related to the information contents as well as to the specific language, as the choice of the words; (3) In the surface realization, sentences are finally generated on the base of the decision taken by the previous modules and by considering the constraints related to the language specific word order and inflections.

For our system, the contents of information that have to be communicated, i.e. the document planning, are produced by the reasoner. Moreover, with the aim to easily implement in the messages the prescriptions of the persuasion theories, we adopted the simplest architecture for NLG. We treat sentence planning and surface realization in one single module by adopting a *template-based* approach. We use five templates to communicate the five cases of output of the reasoner: in Table 1 we report the cases obtained by the interpretation of the output of the reasoner (column **C**), the direction of the deviation (column **D**), the Italian templates and their rough English translation.

Indeed, the final message is obtained by modifying the templates on the basis of the specific values for the motivation of inconsistency that can be extracted by

| C | D | Message Template | Translation |
|---|---|---|---|
| I.1 | IPO | Questo piatto non va affatto bene, contiene davvero pochissime proteine! | This dish is not good at all, it's too poor in proteins! |
| I.2 | IPO | Ora non puoi mangiare questo piatto perché è poco proteico. Ma se domenica mangi un bel piatto di fagioli allora lunedì potrai mangiarlo. | You cannot have this dish now because it doesn't provide enough proteins, but if you eat a nice dish of beans on Sunday, you can have it on Monday. |
| C.1 | IPO | Va bene mangiare le patatine ma nei prossimi giorni dovrai mangiare più proteine. | It's OK to eat chips but in the next days you'll have to eat more proteins. |
| C.2 | IPO | Questo piatto va bene, è solo un po' scarso di proteine. Nei prossimi giorni anche fagioli però! :) | This dish is OK, but it's a bit poor in proteins. In the next days you'll need beans too! :) |
| C.3 | - | Ottima scelta! Questo piatto è perfetto per la tua dieta :) | Great choice! This dish is perfect for your diet :) |

**Table 1.** The persuasive message templates: the underline denotes the variable parts of the template. The column **C** contains the classification produced by the STP reasoner, while the column **D** contains the direction of the deviation: *IPO* (*IPER*) stands for the information that the dish is poor (rich) in the value of the macronutrient.

interpreting the output of the reasoner (cf. Section 2.3) and possible suggestions that can guide the choices of the user in the next days. The suggestions can be obtained by a simple table that couples the excess (deficiency) of a macronutrient with a dish that could compensate this excess (deficiency). In particular, for the reasoner's outputs I.1, I.2, C.1 and C.2, we need to distinguish the case of a dish poor in a macronutrient (*IPO* in Table 1) with respect to the case of a dish rich in a macronutrient (*IPER*). If the dish is classified as IPO (IPER), we insert into the message a suggestion to consume in the next days a dish that contains a big (small) quantity of that specific macronutrient.

For sake of simplicity we do not describe the algorithm used in the generation module to combine the three distinct outputs of the reasoner on the three distinct macronutrients (i.e. proteins, lipids and carbohydrates). In short, the messages corresponding to each macronutrient need to be *aggregated* into a single message. A number of constraints related to coordination and relative clauses need to be accounted for [22]. In the next Section we describe the three theories of persuasion that influenced and motivated the design of the messages.

### 3.2 Designing persuasive messages in the diet domain

A number of theories on the design of persuasive textual and multimedial messages have been proposed in the last years [14, 23, 10, 21, 7, 11, 12, 16]. Most of these theories can be split in two narrow categories. The first category includes the theories that approach the persuasion from a practical and empirical point of view, by using strategies and methods typical of the psychology and of the interaction design. The second category includes the theories that approach the persuasion from a theoretical point of view, by using strategies and methods typical of strong artificial intelligence and cognitive science. We discuss the three theories that mostly influenced the design of the messages in relation to our project.

CAPTology (Computers As Persuasive Technologies) is the study of computers as persuasive technologies, i.e. "[...] the design, research, and analysis of interactive computing products (computers, mobile phones, websites, wireless technologies, mobile applications, video games, etc.) created for the purpose of changing people's attitudes or behaviors" [10]. The starting point of Fogg's theory is that the computer is perceived by users in three coexisting forms, Tool-Media-SocialActor, and each one of these three forms can exercise some forms of persuasion. As a tool, the computer can enhance the capabilities of a user: our system calculates the nutritional contents of the food, and so it enhances the ability to correctly judge the compatibility of a dish with a diet. As a media the computer "provides experience": in our system, the human memory is enhanced by the reasoner, which indirectly reminds her/him what s/he ate in the last days. As a social actor the computer creates an empathic relationship with the user reminding her/him the "social rules": in our system the messages guide the user towards the choice of a balanced meal, convincing her/him to follow the diet that her/himself decided. Fogg recently defined a number of rules to design effectively persuasive systems [11], and some of these rules have modeled our messages. For example, the rule: *Learn what is Preventing the target behavior*, proposes to classify an "uncorrect" behavior along three major lines: (1) lack of motivation, (2) lack of ability, (3) lack of a well-timed trigger to perform the behavior. In our system all the three components play a role. Indeed, a user follows a bad diet because (i) s/he is not enough motivated, (ii) because s/he does not know that the dish is in contrast to her/his diet (iii) because s/he does not have the right stimulus at the time of choosing a dish. The reasoning and the generated messages are working on the last two components: the reasoner enhances the user's abilities allowing her/him to have the relevant information at the right time, the generation system creates a stimulus (the message) when it is really necessary, *kairos* in the Fogg's terminology, i.e. when the user has to decide what to eat.

Another approach to computational persuasion is strongly related to the concept of *tailoring*, i.e. the adaptation of the output of the computation to a specific user. A pioneering work for tailoring in the field of NLG is described in [21]: the authors have designed an NLG system, called *STOP*, to build a letter that induces a specific reader to quit smoking. The key component of STOP is the individuation of a *user type* by using the answers given to a questionnaire. In this way, one can build a specific user profile. By using this profile the system generates a tailored letter on the basis of a template. This simple approach to persuasion unfortunately did not yield the desired results. The experimental protocol has shown, through the use of a control group, that the enhancement given by customization was negligible. At this stage, we do not adopt in our system the ability to create custom messages for a specific user, but, as evidenced by similar experiences, customization of the feedback could improve the performances of the system. A system for tailoring that we partially adopt in our messages is described in [16], where a series of messages are sent via SMS to reduce the consumption of snacks. In this case, the messages adopt six patterns/templates for persuasion derived from the general theory of persuasion of Cialdini [7]. The six patterns are: (1) Reciprocity: *people feel obligated to return a favor*, (2) Scarcity: *people will value scarce products*, (3) Authority: *people value the opinion of experts*, (4) Consistency: *people do as they said they would*, (5) Consensus: *people do as other people do*, (6) Liking: *we say yes to people we like.* Compared to this classification, all the messages of our generator belong to the patterns of authority and consistency.

One approach to persuasion strictly related to strong artificial intelligence and cognitive science is based on the concept of the computer as an intelligent agent [14, 23, 12]. The system behaves as a real autonomous entity and it is often modeled as a BDI (Beliefs, Desires, Intentions) agent, whose main purpose is to persuade the user to behave in a specific way. This approach has been adopted essentially for research purposes rather than for commercial applications. In contrast to the design of our NL generator, where there is a single module based on templates, such agent-based approach allows a great modularity in the design of a persuasive system. We describe some issues of these systems in order to understand the deficiencies of our simple approach. Hovy defines a number of heuristic rules that constrain the “argument” defined in the process of sentence planning. For example: *Adverbial stress words can only be used to enhance or mitigate expressions that carry some affect already* [14]. In a similar way, De Rosis and Grasso define a number of heuristic rules on the argument structure, to lexically enhance or mitigate a message [23]. The use of certain adverbs, as little bit (*poco*), very (*molto*), really (*davvero*), are used to enhance some specific argument structures. Indeed, we adopt this strategy by using this kind of adverbs in the messages I.1, I.2, C.1 and C.2. Guerini et al. define a detailed taxonomy of persuasion strategies that a system can adopt and relate the strategies to the theory of argumentation [12]. Moreover, they define an architecture for persuasion that follows the standard modularization of NLG systems. This allows for a very rich persuasive action, which begins from the planning of a rhetorical structure in the content planning. Compared to the taxonomy of the proposed strategies, we can see that our messages belong to one single category, called action_inducement/goal_balance/positive_consequence. This strategy induces an action (to choose a dish), by using the user’s goal (a balanced diet) and by using the benefits deriving from this goal.

Finally, note that in the messages C.2 and C.3 we used emoticons. Indeed, some studies showed that the use of emoticons in written texts can increase the communicative strength of a message. For example Dirke shows that the use of emoticons sets a tone of friendship to the message type and can increase the positive value of the message [9].

## 4 Conclusions and Related works

There are a number of academic studies that are related to our project, among them [3, 15], and there is also a great number of smartphone applications related to nutrition, e.g. *DailyBurn, Lose It!, MyNetDiary, A low GI Diet, Weight-*

*Watchers*. However, our dietary system presents two elements of novelty: (1) the use of automatic reasoning as a tool for verifying the compatibility of a specific recipe with a specific diet and for determining the consequences of the choice of a specific dish and (2) the use of NLG techniques to produce the answer.

Some authors have applied Operational Research techniques to tackle the problem of planning a diet (see the survey in [18] or the more recent paper [5]). These techniques are based on the simplex method for solving linear programming problems. However these approaches are meant to plan an entire diet and they do not support the user in choosing a dish and in investigating the consequences of her/his choice. In [6] the authors have tackled the problem of assessing the compatibility of a single meal to a norm and of suggesting to the user some actions to balance the meal (e.g. removing/adding food); they employed fuzzy arithmetic to represent imprecision/uncertainty in quantity and composition of food and heuristic search for determining the actions to be suggested. They did not consider the problem of globally balancing the meals.

In the next future, we intend to improve the NLG module for tailoring. In particular, we want 1) to build a corpus of sentences that a professional dietician would use to persuade users towards correct dish choices, 2) to separate microplanning from realization, 3) to classify the users in types in order to personalize the messages on the basis, for instance, of the age. Finally, we plan to experiment the system in two settings. First we intend to design a simulation that includes 1) a database of real recipes, 2) a user model that allows to test the persuasion efficacy and 3) a baseline built rigidly sticking to DRVs. Second, we intend to test the system with a focus group in a clinical setting, in particular with patients affected by essential obesity. In this setting we imagine that the system could be used also by human dieticians for the supervision of their patients.

## References

1. LARN - Livelli di Assunzione di Riferimento di Nutrienti ed energia per la popolazione italiana - IV Revisione. SICS Editore, Milan, Italy (2014)
2. Anselma, L., Terenziani, P., Montani, S., Bottrighi, A.: Towards a comprehensive treatment of repetitions, periodicity and temporal constraints in clinical guidelines. Artificial Intelligence in Medicine 38(2), 171 – 195 (2006)
3. Balintfy, J.L.: Menu planning by computer. Commun. ACM 7(4), 255–259 (1964)
4. Barzilay, R., Mccullough, D., Rambow, O., Decristofaro, J., Korelsky, T., Lavoie, B., Inc, C.: A new approach to expert system explanations. In: 9th International Workshop on Natural Language Generation. pp. 78–87 (1998)
5. Bas, E.: A robust optimization approach to diet problem with overall glycemic load as objective function. Applied Mathematical Modelling 38(19–20), 4926 – 4940 (2014)
6. Buisson, J.C.: Nutri-educ, a nutrition software application for balancing meals, using fuzzy arithmetic and heuristic search algorithms. Artif. Intell. Med. 42(3), 213–227 (Mar 2008)
7. Cialdini, R.B.: Influence: science and practice. Pearson Education, Boston (2009)

8. Dechter, R., Meiri, I., Pearl, J.: Temporal constraint networks. Artif. Intell. 49(1-3), 61–95 (May 1991)
9. Derks, D., Bos, A.E.R., von Grumbkow, J.: Emoticons in computer-mediated communication: Social motives and social context. Cyberpsy., Behavior, and Soc. Networking 11(1), 99–101 (2008)
10. Fogg, B.: Persuasive Technology. Using computers to change what we think and do. Morgan Kaufmann Publishers, Elsevier, San Francisco (2002)
11. Fogg, B.: The new rules of persuasion (2009), `http://captology.stanford.edu/resources/article-new-rules-of-persuasion.html`
12. Guerini, M., Stock, O., Zancanaro, M.: A taxonomy of strategies for multimodal persuasive message generation. Applied Artificial Intelligence 21(2), 99–136 (2007)
13. Holzinger, A., Dorner, S., Födinger, M., Valdez, A.C., Ziefle, M.: Chances of increasing youth health awareness through mobile wellness applications. In: Proc. of the 6th Int. Conf. on HCI in Work and Learning, Life and Leisure: Workgroup Human-computer Interaction and Usability Engineering. pp. 71–81. USAB'10, Springer-Verlag (2010)
14. Hovy, E.H.: Generating Natural Language Under Pragmatic Constraints. Lawrence Erlbaum, Hillsdale, NJ (1988)
15. Iizuka, K., Okawada, T., Matsuyama, K., Kurihashi, S., Iizuka, Y.: Food menu selection support system: considering constraint conditions for safe dietary life. In: Proceedings of the ACM multimedia 2012 workshop on Multimedia for cooking and eating activities. pp. 53–58. CEA '12, ACM, New York, NY, USA (2012)
16. 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. TiiS 2(2), 10 (2012)
17. Lacave, C., Diez, F.J.: A review of explanation methods for heuristic expert systems. Knowl. Eng. Rev. 19(2), 133–146 (Jun 2004)
18. Lancaster, L.M.: The history of the application of mathematical programming to menu planning. European Journal of Operational Research 57(3), 339 – 347 (1992)
19. Montanari, U.: Networks of constraints: Fundamental properties and applications to picture processing. Information sciences 7, 95–132 (1974)
20. Nishida, C., Uauy, R., Kumanyika, S., Shetty, P.: The joint WHO/FAO expert consultation on diet, nutrition and the prevention of chronic diseases: process, product and policy implications. Public Health Nutrition 7, 245–250 (1 2004)
21. Reiter, E., Robertson, R., Osman, L.: Lessons from a Failure: Generating Tailored Smoking Cessation Letters. Artificial Intelligence 144, 41–58 (2003)
22. Reiter, E., Dale, R.: Building Natural Language Generation Systems. Cambridge University Press, New York, NY, USA (2000)
23. de Rosis, F., Grasso, F.: Affective interactions. chap. Affective Natural Language Generation, pp. 204–218. Springer-Verlag, New York, NY, USA (2000)
24. Schofield, W.N.: Predicting basal metabolic rate, new standards and review of previous work. Human Nutrition: Clinical Nutrition 39C, 5–41 (1985)
25. Terenziani, P., Anselma, L.: A knowledge server for reasoning about temporal constraints between classes and instances of events. International Journal of Intelligent Systems 19(10), 919–947 (2004)
26. World Health Organization: Global strategy on diet, physical activity and health (WHA57.17). In: 75th World Health Assembly (2004)