An Approach for Explaining Reasoning on the Diet Domain

Luca Anselma, Alessandro Mazzei

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

Abstract. We are conducting a project involving automatic reasoning and natural language generation in the domain of diet management. In this paper we describe the main issues related to the automatic reasoning mechanisms for diet management purposes and we present the message generation techniques designed to support the users in managing their dietary choices. In particular, this paper has two main goals. First, with respect to the automatic reasoning module, we briefly describe the main features of the Simple Temporal Problem formalization of the diet domain. Second, with respect to the natural language generation module, we describe the actual implementation of the tasks of data interpretation and content selection, and the design of the message templates; moreover, we give a novel formalization of the sentence aggregation algorithm.

1 Introduction

People might fail to follow a healthy diet for a number of reasons. Sometimes they do not know that a dish is in contrast to their diet or, in other cases, they are not motivated enough since they do not have the right stimulus at the right time. So, a diet management system needs to *reason* in order to enhance the users' computational abilities to recognize healthy dishes and needs to *generate* a persuasive stimulus when it is really necessary, i.e., when the users have to decide what to eat.

Similarly to [21] and [8] we want to investigate the possibility to apply automatic reasoning and persuasive NLG on mobile devices for helping people in reaching a virtuous behaviour. In this paper we address some issues related to numerical reasoning and persuasive natural language generation in the diet domain. We describe the actual implementation of the reasoning and generation modules of the diet management system called MADiMan (Multimedia Application for Diet Management) [5, 25]. In particular, after a brief description of the numerical reasoner, which has been extensively described in [3, 6], we provide a detailed description of the generation module.

The MADiMan system is a virtual dietitian (Fig. 1) designed: (1) to recover the nutritional information directly from a specific recipe, (2) to reason over recipes and diets by allowing some forms of diet disobedience, and (3) to persuade the user to minimize these acts of disobedience. MADiMan offers facilities

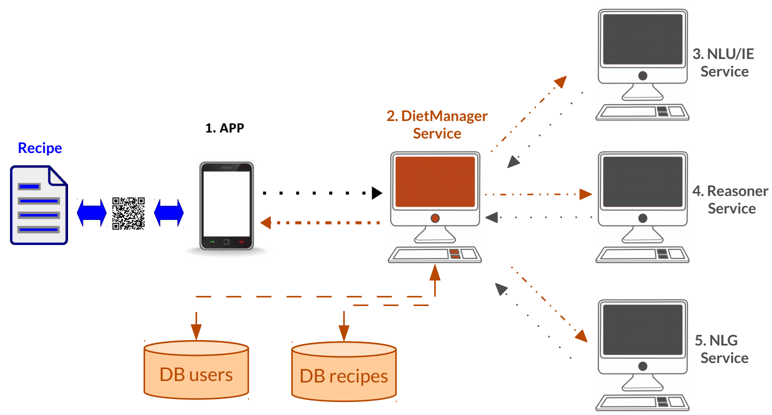

Fig. 1. The architecture of the diet management system.

to check the compatibility and to foresee the impact that a specific meal has on a specific diet. In a diet it is necessary to consider parameters such as energy requirements and the amount of macronutrients such as carbohydrates, lipids and proteins. The medical literature (e.g., [23]) provides Dietary Reference Values (DRVs) that can be computed from user information such as weight, gender, age, lifestyle. In MADiMan the reasoning module is a numerical reasoner based on Simple Temporal Problems (STPs) where we represent the DRVs as STP constraints [11] by substituting the temporal distance with the DRVs and the caloric values of a dish distributed on the three macronutrients [6]. By using the ideal value for calories, MADiMan evaluates the compatibility of the specific dish with the actual status of the diet. Moreover, in order to provide a user-friendly information not limited to "consistent/inconsistent" answer and to make it also useful for the sake of user persuasion, MADiMan converts the numerical reasoning into a symbolic form that is suitable for the generation of NL messages [31].

In the applicative scenario, the interaction between the man and the food is mediated by an artificial intelligent system that encourages or discourages the user to eat a meal that s/he intends to eat. The main factors that the system needs to manage are: (1) the diet that the user intends to follow, (2) the food that has been eaten in the last days, and (3) the specific recipes of the dishes and the nutritional values of their ingredients. The complete MADiMan system is composed of five modules (see Fig. 1): a smartphone application (app), a central module that manages the information flow (DietManager), a Natural Language Understanding / Information Extraction (NLU/IE) module, a reasoning module (Reasoner), a natural language generator module (NLGenerator). Upon a registration where the user provides her anthropometric data, the information flow follows this pattern:

1. The user, by using the app, recovers the QR-code of a specific recipe, or finds the recipe using full-text search.
2. The app, using the DietManager service, retrieves the user diet together with the list of the food that the user has eaten in the last days. Moreover, the DietManager retrieves the specific recipe of the specific dish.
3. The NLU/IE module computes the salient nutrition information about the specific dish.
4. The Reasoner, using the user diet and the list of the food that has been eaten in the last days, produces a judgment about the dish for the user.
5. The NLGenerator uses the result given by the Reasoner, produces an explanation for the user in plain natural language and/or in a multimedial format (e.g. by using icons).
6. The DietManager sends the result produced by the NLGenerator to the app: the user will see this final result on her smartphone. If the user decides to eat the dish, the app will send this information to the DietManager that will update the list of food eaten. So, the DietManager plays the role of a hub.

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. However, for a complete implementation, in the diet management architecture depicted in Figure 1 there are a number of choices which need to be experimentally evaluated. For instance, there are many technologies available on smartphones, such as the GPS localization or speech recognition, that could be used for the insertion of the data. All these choices are important to create a highly usable system supporting and motivating users in their daily life [24].

In this paper[1], we first briefly describe the STP reasoner in Section 2, and second we give some details concerning the NLG module in Section 3. In particular we describe current implementation of three specific modules of the generator: in Section 3.1, we describe the data interpretation and content selection process for converting the numerical output of the STP reasoner into a symbolic form. In Section 3.2, we describe the design of five templates for the realization of the persuasive messages for the three macronutrients, and in Section 3.3 we provide a novel formalization of the algorithm used to aggregate the messages. Finally, Section 4 closes the paper with some conclusions and pointers to future work.

[1] Some details on the app, the DietManager service, the NLU/IE service can be found on the webpage of the project MADiMan: `http://di.unito.it/madiman`.

2 Automatically reasoning on diets

In this section first we introduce "Simple Temporal Problem" (STP), and then we describe how we exploit STP for representing reasoning over dietary information. In contrast to other approaches to diet formalization, for example [15][2] which is based on ontologies, we approach the diet management problem from a numerical perspective since STP is a numerical constraint satisfaction formalism.

2.1 Simple Temporal Problems

An *STP constraint* [11] is 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 whose domain can be either discrete or real. An STP constraint can be interpreted in the following way: the temporal distance between the temporal points x and y must be between c – the lower bound of the distance – and d – the upper bound. It is also possible to impose strict inequalities (i.e., $<$) and to use $-\infty$ and $+\infty$ to denote the fact that there is no lower or upper bound, respectively. An *STP* is a conjunction of STP constraints.

STP can be represented as a graph whose nodes are the temporal points and the arcs are labeled with the temporal distance between the points.

A prominent feature of STP is that the problem of determining the *consistency* of an STP is tractable and the algorithm employed, i.e., an all-pairs shortest paths algorithm such as Floyd-Warshall's one, also obtains the minimal network. A *minimal network* is equivalent to the original STP and it makes explicit all the implied STP constraints by representing the strictest constraint between each pair of point, which corresponds to the minimum and maximum distance between each pair of points. 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 [11]. Its temporal computational cost is cubic in the number of the temporal points.

2.2 Reasoning on the energy requirements and energy intake

It has been proven that excessive calorie intake – and subsequent obesity – increases the risk of developing chronic disease and decreases life expectancy [17]. In order to restrict calorie intake, one of the most important feature to be taken into account in a diet is the total energy requirement. In particular, from a user's weight, gender and age, using Schofield equation [34], it is possible to estimate her basal metabolic rate as reported in Table 1. For example a 40-year-old male who weighs 71.3 kg has an estimated basal metabolic rate of 1690 kcal/day. Such value is then adjusted by multiplying the basal metabolic rate by a factor related with the physical activity of the individual; for example, if the person has a sedentary lifestyle, a factor of 1.45 is employed so that he has a total energy requirement of $1690 \cdot 1.45 = 2450$ kcal/day. It is worth noting that our

[2] We thank an anonymous reviewer for pointing us to this work.

system does not depend on the particular formula used to compute the total energy requirement, and it is possible to employ other formulas or the dietitian can even empirically estimate the value for specific patients.

Using STP constraints it is possible to represent the total energy requirement and also the dietary prescription regarding the meals' calorie amounts. For instance, the daily energy requirement of 2450 kcal/day is represented by the STP constraint $day_E - day_S = 2450$ and the 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 day_E, day_S, $lunch_E$ and $lunch_S$ represent the end and the start of the day and of the lunch, respectively.

2.3 Supporting dietary transgressions

We exploit the STP framework to allow users to make small deviations from the ideal goal of sticking to their energy requirements and to know in advance what are the consequences of such deviations on the rest of the diet.

Thus, we admit a tolerance in adhering to the diet constraints. Such a tolerance generates loose constraints over the shortest periods (i.e., days and meals) and strict constraints over the longest periods (i.e., weeks), thus allowing episodes of transgression in the short term that however do not prevent a user from reaching the diet goal. Let σ_w, σ_d and σ_m be the tolerances for the week, the days and the meals respectively, $\sigma_w \leq \sigma_d \leq \sigma_m$. For example the recommended energy requirement of 2450 kcal/day, considered over a week and considering a tolerance σ_w, results in a constraint such as $2450 \cdot 7 \cdot (1 - \sigma_w) \leq week_E - week_S \leq 2450 \cdot 7 \cdot (1 + \sigma_w)$ and for the single days we allow the user to have a tolerance σ_d, thus resulting in the constraints $2450 \cdot (1 - \sigma_d) \leq Sunday_E - Sunday_S \leq 2450 \cdot (1 + \sigma_d)$, $\ldots$, $2450 \cdot (1 - \sigma_d) \leq Saturday_E - Saturday_S \leq 2450 \cdot (1 + \sigma_d)$ (see Fig. 2a).

For single meals we consider that the energy assumption for the day is split among the meals; for example, assigning 20% for breakfast, 40% for lunch and 40% for dinner, and that we allow a tolerance of σ_m, we obtain constraints such as $2450 \cdot 20\% \cdot (1 - \sigma_m) \leq Sunday\_breakfast_E - Sunday\_breakfast_S \leq 2450 \cdot 20\% \cdot (1 + \sigma_m)$ and $2450 \cdot 20\% \cdot (1 - \sigma_m) \leq Sunday\_lunch_E - Sunday\_lunch_S \leq 2450 \cdot 20\% \cdot (1 + \sigma_m)$.

Age (years)	**Basal metabolic rate (kcal/day)**
Men	
$18 - 29$	$((0.063 \cdot weight) + 2.896) \cdot 238.8459$
$30 - 59$	$((0.048 \cdot weight) + 3.653) \cdot 238.8459$
> 60	$((0.049 \cdot weight) + 2.459) \cdot 238.8459$
Women	
$18 - 29$	$((0.062 \cdot weight) + 2.036) \cdot 238.8459$
$30 - 59$	$((0.034 \cdot weight) + 3.538) \cdot 238.8459$
> 60	$((0.038 \cdot weight) + 2.755) \cdot 238.8459$

Table 1. Schofield equation for estimating the basal metabolic rate [34].

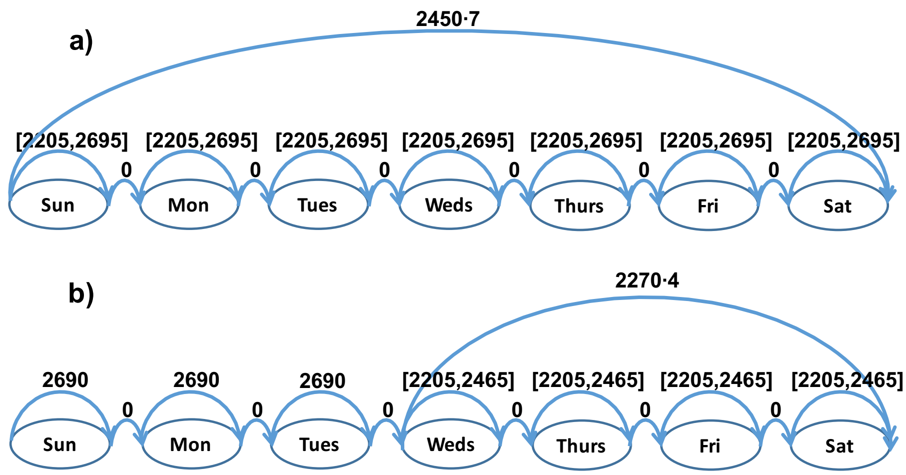

Fig. 2. a) Example of STP constraints for a week (with $\sigma_w = 0$ and $\sigma_d = 10\%$). For space constraints the constraints for the meals are not represented. **b)** Example of STP constraints after three day when the user has eaten 2690 kcal and the constraint propagation has been performed.

In order to build a sound STP, it is also necessary to add STP constraints such the ones that impose that a day ends when the subsequent day starts (see, for example, the constraints 0 in Fig. 2a), that a week starts when the first day of the week starts, that a week ends when the last day of the week ends and the analogous constraints for the meals.

A *compatible* meal is a meal that satisfies all the STP constraints that involve such a meal.

2.4 Representing and reasoning on the diet and the food

We wish to support users into taking advantage of the information regarding the actual meals they consume. In this way, such users can apprehend what are the consequences on their diet of eating a specific meal and they could use such information to make informed decisions about their current or future meals. Therefore it is necessary to "integrate" the information about the eaten meals with the dietary recommendations. Thus, as the users eat their meals, we substitute in the STP the original STP constraint related with the diet with the information regarding the actual energetic supply of such meals.

For example, let us suppose that a 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 $Sunday_E - Sunday_S = 2690$, $\ldots$, $Tuesday_E - Tuesday_S = 2690$. Then, propagating the constraints of the new STP, we discover (see Fig. 2b) 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.

3 Persuasive NLG for diet

A widespread architecture for NL generation, especially in applicative contexts, is a pipeline composed by three distinct modules: the document planning, the sentence planning and the surface realization [32]. Each one of these modules addresses distinct issues, in particular: (1) in the document planning one decides what to say, i.e., which information contents will be communicated; (2) in the sentence 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 STP reasoner, and we need to convert such contents into a human-readable form: we address this issue in the Section 3.1. Moreover, with the aim to easily implement in the messages the prescriptions of the persuasion theories, we adopted the simplest architecture for NLG. Similar to other works, e.g. [15], we treat sentence planning and surface realization in one single module by adopting a *template-based* approach. We describe the templates in the Section 3.2. Finally, in Section 3.3 we formalize the sentence aggregation algorithm used to merge messages concerning distinct macronutrients.

3.1 Data interpretation and content selection

In order to show to the user a meaningful feedback, it is necessary to interpret the data resulting from the STP. Here we consider the case where the user proposes to the system a dish, the system obtains its caloric value, translates it along with the user's diet and past meals into an STP and, by propagating the constraints, obtains the minimal network. For sake of clarity, we present the content selection algorithm by considering one single generic macronutrient, but the real suitability of a dish depends on the results of the three macronutrients (see Section 3.3).

Using the resulting STP it is possible to classify 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_1) and *consistent and perfectly balanced* (C_3). In the cases I_1 and I_2 the energy supply of the dish is inconsistent. In case I_1 the energy supply is inconsistent with regard to the user's diet as represented in the STP considering the tolerance values. The dish cannot be accepted even independently of the other food the user may possibly eat. This case is detected by considering whether the nutritional value of the dish violates a constraint in the STP. In case I_2 the dish per se does not violate the diet constraints, but – considering the past meals the user has eaten – it would preclude to be consistent with the diet. Thus, it is inconsistent now, but in the future, e.g., next week, it could become possible to choose it. This case is detected by determining whether the energy supply, despite it satisfies

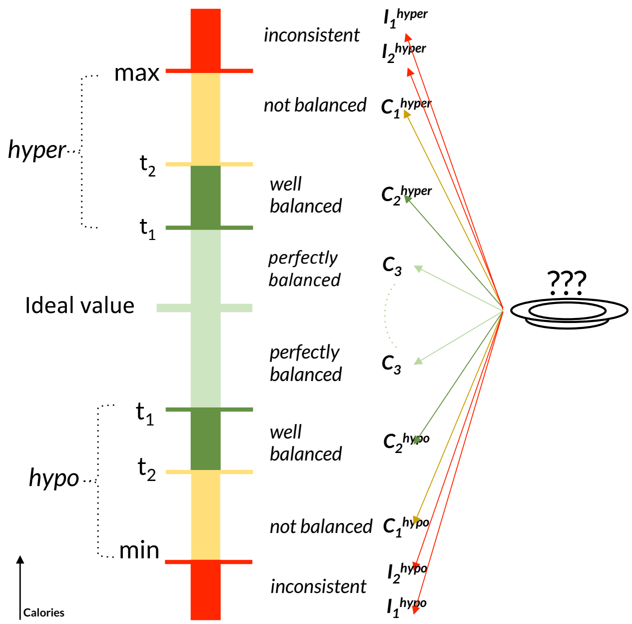

Fig. 3. Classification of an inconsistent/consistent value of a meal's energy supply given the minimum and maximum value of an STP constraint.

the constraints in the initial STP, is inconsistent with the STP that contains also the constraints related to the food that the user has actually eaten so far.

In the cases C_1, C_2 and C_3 the value of the energy supply is consistent with the diet, also taking into account the other meals that the user has already eaten. It is possible to detect that a meal is consistent by exploiting the minimal network of the STP: if the value of the energy supply is included between the lower and upper bounds of the relative STP constraint, then the STP is certainly consistent and the meal is consistent with the diet. A consistent but not balanced choice of a meal will have consequences on the rest of the user's diet because the user will have to "compensate" it. Thus we distinguish three cases depending on the level of the adequacy to the diet of the meal's energy supply. In order to discriminate between the cases C_1, C_2 and C_3, we consider how the value of the energy supply stacks upon the allowed range represented in the related STP constraint. We assume that the mean value is the "ideal" value according to the diet's goals and we consider two parametric user-adjustable thresholds relative to the mean: we classify the meal according to the distance from the ideal value as not balanced (C_1), well balanced (C_2) or perfectly balanced (C_2) (see Fig. 3). In particular, we distinguish between excess or lack of energy supply for a meal: if a meal is in excess (lacking) with regard to the ideal value, we tag the meal with the keyword *hyper* (*hypo*). This information is exploited in the generation of the messages.

3.2 Designing five templates for realizing persuasive messages

In order to convert in sentences the five possible kinds of output of the STP reasoner, at this stage of the project we adopted a simple template-based realizer that produces five kinds of messages designed for persuasion. We use five templates to communicate the five cases of output of the reasoner, that are I_1, I_2, C_1, C_2 and C_3. For sake of clarity, we describe the templates by assuming one single macronutrient. In the Section 3.3 we discuss how to aggregate the three messages generated for the three distinct macronutrients.

In the NLG community there are a few works treating the generation of persuasive messages the diet domain, e.g. [27, 15]. Moreover, a larger number of works considering the application of NLG for presenting the results of automated reasoning to the user, especially in the case of expert systems, e.g. [36, 7, 22]. Moreover, a number of theories on the design of persuasive textual and multimedial messages have been proposed in the last years. 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 [16, 30, 9, 21]. 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 [20, 33, 19].

As a working hypothesis, the actual implementation of the generator produces messages following a fixed rhetorical schema. So, the final message will be composed by three parts: a global evaluation of the dish, three evaluations for the macronutrients (i.e. carbohydrates, lipids, proteins), and a suggestion on the future dishes to eat.

In Table 2, we report the five cases obtained by the interpretation of the output of the reasoner and the direction of the deviation (column **O**), and the template[3] (column **Message Template**). 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 interpreting the output of the reasoner (cf. Section 3.1) and some possible suggestions that can guide the choices of the user in the next days. The suggestions can be obtained by querying a database in order to couple the excess (deficiency) of a macronutrient with a dish that can 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 low in a macronutrient (*hypo* in Table 2) with respect to the case of a dish high in a macronutrient (*hyper*). If the dish is classified as *hypo* (*hyper*), we insert into the message a suggestion to consume in the next days a dish with an increased (reduced) amount of that macronutrient.

Adopting persuasion theories in the templates

[3] For space reasons, we report here the English version of the message and we omit the original Italian version.

O	Message Template
I_1^{hypo}	This dish is not good at all, it's too poor in proteins!
I_2^{hypo}	You cannot have this dish now because it doesn't provide enough proteins, but if you eat a nice dish of beans Sunday, you can have it on Monday.
C_1^{hypo}	This dish is good, but it's poor in proteins. In the next days you'll have to eat really more proteins.
C_2^{hypo}	This dish is OK, but it's a bit poor in proteins. In the next days you'll need more proteins! :)
C_3	Great choice! This dish is perfect for the proteins in your diet :)

Table 2. The persuasive message templates for the STP modeling caloric value for the proteins. The underlined text vary among the three macronutrients.

Similar to [21], the Cialdini's general theory of persuasion has inspired our design of the messages [9]. Cialdini states that there are six patterns which are characteristic of human nature: (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.* Note that compared to the six Cialdini's persuasion patterns, all the messages in Table 2 belong to the patterns of authority and consistency.

With respect to the low-level linguistic strategies, by following [33], we used a number of adverbs (*little bit, very, really*) in order to enhance or mitigate a message for the cases I_1, I_2, C_1 and C_2. Furthermore, compared to Guerini et al. persuasive strategies taxonomy [19], we can see that all the messages belong to one single category, called *action-inducement & goal-balance & positive-consequence*. This strategy induces an action (i.e. *to choose a dish*), by using the user's goal (i.e. *a healthy 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 have showed that the use of emoticons in written texts can increase the communicative strength of a specific class of messages. In particular, Dirke has showed that the use of emoticons sets a tone of friendship to the communication and can increase the positive value of the message [12].

3.3 Modelling aggregation for messages on macronutrients

The aggregation plays an important role to generate fluent and efficient texts [13]. In the previous section we presented a template-based approach for the generation of a message concerning one single macronutrient. In this section we introduce an algorithm for aggregating the three messages for the three macronutrients. We write (O_C, O_L, O_P) to indicate the symbolic output for carbohydrates, lipids and proteins respectively, where

$$O_X \in \{I_1^{hypo}, I_1^{hyper}, I_2^{hypo}, I_2^{hyper}\, C_1^{hypo}, C_1^{hyper}, C_2^{hyper}, C_2^{hypo}, C_3\}$$

Indeed, a trivial aggregation strategy could merge only messages that belong to the same category, i.e. $O_x = O_y$: this simple strategy corresponds to the *syntactic aggregation* in the classification of [29]. However, we design an aggregation strategy that accounts for a more sophisticated form of *conceptual aggregation*. The general idea of the aggregation algorithm is to merge messages that are in same category giving more emphasis to the incompatibility messages. So, there are three alternative cases:

A. $\exists X \in \{C, L, P\} : O_X = I_1^{hypo} \vee O_X = I_1^{hyper}$
B. $\forall X \in \{C, L, P\} : (O_X \neq I_1^{hypo} \wedge O_X \neq I_1^{hyper}) \wedge \exists Y \in \{C, L, P\} : (O_Y = I_2^{hypo} \vee O_Y = I_2^{hyper})$
C. *otherwise*, i.e. $\forall X \in \{C, L, P\}\ \exists i \in \{1, 2, 3\} : (O_X = C_i^{hypo} \vee O_X = C_i^{hyper})$

In the case **A.** there is at least a permanent inconsistency for a macronutrient, in the case **B.** there is no permanent inconsistencies but at least an occasional inconsistency, and in the case **C.** all the macronutrients are consistent.

In the cases **A.** and **B.**, we aggregate the messages by exploiting the information about incompatibility, that is (i) by merging the messages about incompatible macronutrients, (ii) by removing the messages concerning the compatible macronutrients; moreover, for the case **B.**, (iii) we give a suggestion for the future meals. For instance, suppose that the symbolic interpretation of the STP gives $(O_C = I_1^{hypo}, O_L = I_1^{hypo}, O_P = C_1^{hyper})$, the final message will be:

You cannot eat this dish because it's too poor in carbohydrates and lipids!

while in the case of $(O_C = I_2^{hypo}, O_L = C_2^{hypo}, O_P = I_2^{hypo})$, the final message will be:

You cannot have this dish now because it doesn't provide enough carbohydrates and proteins, but if you eat a nice dish of pasta and beans on Sunday, you can have it on Monday.

In the case **C.**, that is when the macronutrients are all compatible, we define 10 distinct aggregation patterns (cf. [10]) by considering the possible occurrences of the values C_1, C_2 and C_3 in (O_C, O_L, O_P) disregarding the order (i.e. the specific macronutrient).

In this case the algorithm first aggregates messages belonging to categories C_1 and C_2, and second orders the aggregated messages. The order of the aggregated messages follows their compatibility value: the idea is to start with the most positive feedback, as suggested by some theories of persuasion [35, 14]. Formally, the patterns aggregate two messages $O_x = C_i$ and $O_y = C_j$ if $i, j \in \{1, 2\}$, and the message $O_x = C_i$ will be communicated earlier of $O_y = C_j$ if $i > j$. For instance, in the case of $(O_C = C_1^{hypo}, O_L = C_2^{hypo}, O_P = C_3)$, the aggregated message will be:

This dish is good. It is perfect for the proteins, but it is a bit poor in lipids and really poor in carbohydrates,

while in the case of $(O_C = C_3, O_L = C_2^{hyper}, O_P = C_3)$, the aggregated message will be:

This dish is good. It is perfect for the proteins and the carbohydrates, but it is a bit too rich in lipids.

4 Conclusions and future work

In this paper we have described an approach for diet management based on automatic numerical reasoning and natural language generation. We have introduced the main features of an STP-based reasoning for the diet domain. Moreover, we have illustrated the design of the NL generation of persuasive messages in the diet domain, i.e. content selection, linguistic design of templates and aggregation algorithm.

As a future work, in order to augment the variety of the messages, we intend to replace the template-based realizer with a realization engine [18, 26].

We have some experience on working on patient data and collaborating with hospitals [28, 2, 1, 4] and we envision a collaboration with hospital dietitians to evaluate the system by comparing the persuasive NLG output with graphical output representing the numerical values of the macronutrients. We are currently working on the implementation of a mobile app and on its exploitation in a human evaluation of MADiMan, performed by nutrition and dietistics students which will be supervised by some professional dietitians.

References

1. Anselma, L., Bottrighi, A., Montani, S., Terenziani, P.: Extending bcdm to cope with proposals and evaluations of updates. IEEE Transactions on Knowledge and Data Engineering 25(3), 556–570 (2013)
2. Anselma, L., Bottrighi, A., Montani, S., Terenziani, P.: Managing proposals and evaluations of updates to medical knowledge: Theory and applications. Journal of Biomedical Informatics 46(2), 363–376 (2013)
3. Anselma, L., Mazzei, A., Piovesan, L., De Michieli, F.: Adopting stp for diet management. In: Proceedings of 2014 IEEE International Conference on Healthcare Informatics, ICHI 2014. p. 371 (2014)
4. Anselma, L., Montani, S.: Planning: Supporting and optimizing clinical guidelines execution. vol. 139, pp. 101–120 (2008)
5. Anselma, L., Mazzei, A.: Towards Diet Management with Automatic Reasoning and Persuasive Natural Language Generation. In: Progress in Artificial Intelligence - 17th Portuguese Conference on Artificial Intelligence, EPIA 2015, Coimbra, Portugal, September 8-11, 2015. Proceedings. pp. 79–90 (2015)
6. Anselma, L., Mazzei, A., De Michieli, F.: An artificial intelligence framework for compensating transgressions and its application to diet management. Journal of Biomedical Informatics 68, 58–70 (April 2017)
7. 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)
8. Braun, D., Reiter, E., Siddharthan, A.: Creating textual driver feedback from telemetric data. In: ENLG 2015 - Proceedings of the 15th European Workshop on Natural Language Generation, 10-11 September 2015, University of Brighton, Brighton, UK. pp. 156–165 (2015)
9. Cialdini, R.B.: Influence: science and practice. Pearson Education, Boston (2009)
10. Dalianis, H., Hovy, E.: Aggregation in natural language generation, pp. 88–105. Springer Berlin Heidelberg, Berlin, Heidelberg (1996)

11. Dechter, R., Meiri, I., Pearl, J.: Temporal constraint networks. Artif. Intell. 49(1-3), 61–95 (May 1991)
12. 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)
13. DiEugenio, B., Fossati, D., Yu, D., Haller, S.M., Glass, M.: Aggregation Improves Learning: Experiments in Natural Language Generation for Intelligent Tutoring Systems. In: ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA. pp. 50–57 (2005)
14. Dohrenwend, A.: Serving up the feedback sandwich. Family practice management 9(10), 43–50 (2002)
15. Dragoni, M., Bailoni, T., Eccher, C., Guerini, M., Maimone, R.: A semantic-enabled platform for supporting healthy lifestyles. In: Proceedings of the Symposium on Applied Computing. pp. 315–322. SAC '17, ACM, New York, NY, USA (2017)
16. Fogg, B.: Persuasive Technology. Using computers to change what we think and do. Morgan Kaufmann Publishers, Elsevier (2002)
17. Fontana, L., Klein, S.: Aging, adiposity, and calorie restriction. JAMA 297(9), 986–994 (2007)
18. Gatt, A., Reiter, E.: SimpleNLG: A Realisation Engine for Practical Applications. In: Proceedings of the 12th European Workshop on Natural Language Generation. pp. 90–93. ENLG '09, Association for Computational Linguistics, Stroudsburg, PA, USA (2009)
19. Guerini, M., Stock, O., Zancanaro, M.: A taxonomy of strategies for multimodal persuasive message generation. Applied Artificial Intelligence 21(2), 99–136 (2007)
20. Hovy, E.H.: Generating Natural Language Under Pragmatic Constraints. Lawrence Erlbaum, Hillsdale, NJ (1988)
21. 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)
22. Lacave, C., Diez, F.J.: A review of explanation methods for heuristic expert systems. Knowl. Eng. Rev. 19(2), 133–146 (Jun 2004)
23. LARN: LARN - Livelli di Assunzione di Riferimento di Nutrienti ed energia per la popolazione italiana - IV Revisione. SICS Editore, Italy (2014)
24. Marcus, A.: Mobile Persuasion Design - Changing Behaviour by Combining Persuasion Design with Information Design. Human-Computer Interaction Series, Springer (2015)
25. Mazzei, A., Anselma, L., De Michieli, F., Bolioli, A., Casu, M., Gerbrandy, J., Lunardi, I.: Mobile computing and artificial intelligence for diet management. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 9281, 342–349 (2015)
26. Mazzei, A., Battaglino, C., Bosco, C.: SimpleNLG-IT: adapting SimpleNLG to Italian. In: Proceedings of the 9th International Natural Language Generation conference. pp. 184–192. Association for Computational Linguistics, Edinburgh, UK (September 5-8 2016)
27. Mazzotta, I., de Rosis, F., Carofiglio, V.: Portia: A user-adapted persuasion system in the healthy-eating domain. IEEE Intelligent Systems 22(6), 42–51 (2007)
28. Piovesan, L., Anselma, L., Terenziani, P.: Temporal detection of guideline interactions. pp. 40–50 (2015)
29. Reape, M., Mellish, C.: Just what is aggregation anyway? In: ENLG 1998 - Proceedings of the European Workshop on Natural Language Generation (1998)

30. Reiter, E., Robertson, R., Osman, L.: Lessons from a Failure: Generating Tailored Smoking Cessation Letters. Artificial Intelligence 144, 41–58 (2003)
31. Reiter, E.: An architecture for data-to-text systems. In: Proc. of the 11th European Workshop on Natural Language Generation. pp. 97–104. ENLG '07, Association for Computational Linguistics, Stroudsburg, PA, USA (2007)
32. Reiter, E., Dale, R.: Building Natural Language Generation Systems. Cambridge University Press, New York, NY, USA (2000)
33. de Rosis, F., Grasso, F.: Affective interactions. chap. Affective Natural Language Generation, pp. 204–218. Springer-Verlag, New York, NY, USA (2000)
34. Schofield, W.N.: Predicting basal metabolic rate, new standards and review of previous work. Human Nutrition: Clinical Nutrition 39C, 5–41 (1985)
35. Steelman, L.A., Rutkowski, K.A.: Moderators of employee reactions to negative feedback. Journal of Managerial Psychology 19(1), 6–18 (2004)
36. Weiner, J.L.: Blah, A system which explains its reasoning. Artif. Intell. 15(1-2), 19–48 (1980)