Designing and testing the messages produced by a virtual dietitian

Luca Anselma
Dipartimento di Informatica
Università degli Studi di Torino
C.so Svizzera 185, 10149 Torino, Italy
anselma@di.unito.it

Alessandro Mazzei
Dipartimento di Informatica
Università degli Studi di Torino
C.so Svizzera 185, 10149 Torino, Italy
mazzei@di.unito.it

Abstract

This paper presents a project about the automatic generation of persuasive messages in the context of the diet management. In the first part of the paper we introduce the basic mechanisms related to data interpretation and content selection for a numerical data-to-text generation architecture. In the second part of the paper we discuss a number of factors influencing the design of the messages. In particular, we consider the design of the aggregation procedure. Finally, we present the results of a human-based evaluation concerning this design factor.

1 Introduction

The ubiquity of modern technologies allows computers to communicate anytime anywhere with humans. As a consequence, virtual assistant can give positive stimuli when it is really necessary, *kairos* in the Fogg's terminology (Fogg, 2002). In the context of the diet domain, the crucial moment is when people come into a restaurant and decide which dish or menu to order. Often people do not have a healthy diet since they do not know that a specific dish is in contrast to their diet. So, they do not have the correct information, that is the stimulus, at the right time. As a consequence, a *virtual dietitian*, that is a virtual assistant in the diet domain, needs to provide three specific facilities. First, the assistant needs to reason in order to enhance the users' computational abilities to recognize healthy dishes. Second, it needs to generate a persuasive stimulus when it is really necessary, i.e., when users have to decide what to eat. Third, the assistant has to support the user in devising the consequences of a diet transgression.

In this paper we consider the generation of persuasive natural language messages in the diet domain. We describe the actual implementation of the natural language generation (NLG) module of the diet management system called MADiMan (Multimedia Application for Diet Management) (Anselma and Mazzei, 2015). One of the main goals of this project is to investigate on the possibility to apply persuasive NLG for helping people to have a virtuous behavior (Reiter et al., 2003; Kaptein et al., 2012; Braun et al., 2015, 2018; Conde-Clemente et al., 2018). MADiMan performs numerical computation combining food energetic values with diet requirements and reports the result of the computation by using natural language. A crucial point in this process is the combination of information concerning the different macronutrients in the dish, that are carbohydrates, lipids and proteins.

The specific research questions which we want to investigate on in this paper concern the linguistic shape of the messages produced by MADiMan. As a first step towards the building of a complete persuasive system, we evaluate the appealing of the messages by varying two specific linguistic features, that are the aggregation strategy and the lexical choice procedure. We show the first results of a human-based experimentation, that is semantic aggregation increases the engaging of the messages. Moreover, we report some results on the desirability of lexical variability in the messages.

The paper is organized as follows. In Section 2 we give a brief introduction to MADiMan project. In Section 3, we describe the data interpretation and content selection process for converting the numerical output of the numeric reasoner into a symbolic form. In Section 4, we describe the design of the messages that are produced with a realization engine. In particular, in Section 4.1 we discuss two specific algorithms used to aggregate the

Proceedings of The 11th International Natural Language Generation Conference, pages 244–253,
Tilburg, The Netherlands, November 5-8, 2018.

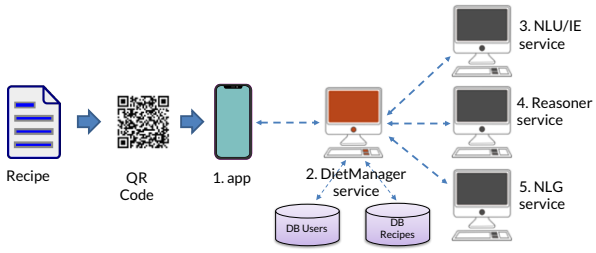

Figure 1: A schema of the MADiMan architecture.

messages. In Section 5, we discuss the experimental setting that we use to give a first human-based evaluation of the message generator. Finally, Section 6 closes the paper with some discussions and pointing on future work.

2 The MADiMan Architecture

The MADiMan system is a virtual dietitian 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 to check the compatibility and to foresee the impact that a specific meal has with a specific diet.

In Figure 1 we depict the architecture of the system implementing the MADiMan virtual dietitian. The information flow is: (1) A user, by using an app, recovers the specific recipe of a dish which she wants to eat. (2) The app, communicating with the DietManager service, retrieves the user diet together with the list of the food that the user has eaten in the last days. (3) The NLU/IE module computes the salient nutrition information about the specific course. (4) The Reasoner, using the user diet and the list of the food that has been eaten in the last days, produces the final recommendation about the dish for the user. (5) The NLGenerator uses the recommendation given by the Reasoner, produces an explanation for the user in simple natural language. (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.

The reasoning module is a numeric reasoner based on Simple Temporal Problems (STPs) (Dechter et al., 1991). In a diet it is necessary to consider parameters such as energy requirements and amount of macronutrients. The medical literature (e.g., (LARN, 2014)) provides Dietary Reference Values (DRVs) that can be computed from user information such as weight, gender, age, lifestyle. For example, let us consider a 40-year-old male who is 1.80 m tall, weighs 71.3 kg and has a sedentary lifestyle; such a person has an energy requirement of 2450 kcal/day. Moreover, he is recommended to assume (LARN, 2014), e.g., 260 kcal/day of proteins, 735 kcal/day of lipids and 1455 kcal/day of carbohydrates. In MADiMan we represent the DRVs as STP constraints (Anselma et al., 2017). STP models a set of constraints as a conjunction of bounds on differences. $c \leq x - y \leq d$, i.e., the distance between the time points x and y is within c and d. In our setting, by substituting the temporal distance between temporal points of STP with the DRVs and the caloric values of a dish distributed on the three macronutrients. Thus, e.g., a constraint 500 kcal $\leq lunchE - lunchS \leq$ 600 kcal imposes that the $distance$ between the start and the end of lunch is between 500 and 600 kcal, i.e., that lunch provides 500-600 kcal. Thus, By using the ideal value for calories (see Fig. 2), MADiMan evaluates the compatibility of the specific dish with the actual status 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 numeric reasoning into a symbolic form that is suitable for the generation of NL messages (Reiter, 2007).

In the next sections, we describe the detail of the algorithm designed to convert the numerical computation in symbols and to elaborate these symbols in order to produce messages.

3 Data interpretation: converting numbers into categories

In order to show to the user a meaningful feedback, it is necessary to interpret the data resulting from the STP. 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 4).

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_2) 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 him 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 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_3) (see Fig. 2). In particular, we distinguish between excess or lack

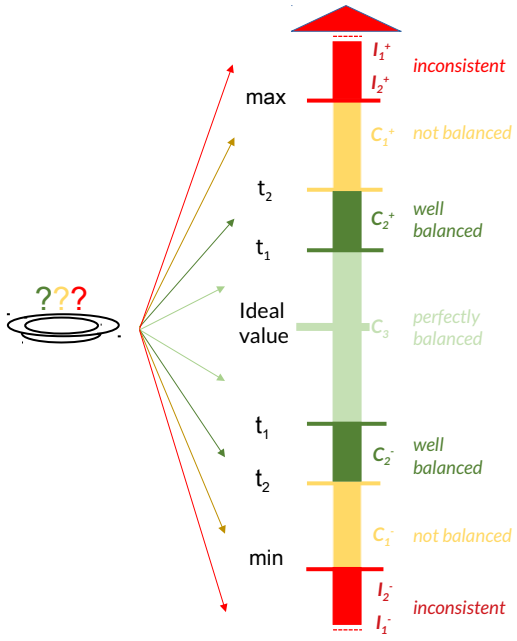

Figure 2: Classification of an inconsistent/consistent value of a meal's energy supply given the minimum and maximum value of an STP constraint.

of energy supply for a meal. If a meal is in excess with regard to the ideal value, we add a $^+$ symbol to the category (e.g. C_2^+) to denote the deviation. In contrast, if a meal is lacking, we add a $^-$ symbol to the category (e.g. C_1^-). This information is exploited in the generation of the messages.

4 Document/Sentence planning and realization

As a working hypothesis, in this stage of the project, MADiMan produces messages following a fixed rhetorical structure and the document plan follows a very simple fixed schema. The final message will be composed by two parts: an overall evaluation of the dish and three evaluations for the macronutrients (i.e., carbohydrates, lipids, proteins)[1]. For the sake of clarity, we now describe the message by assuming one single macronutrient and in Section 4.1 we discuss how to aggregate the three messages generated for the three different macronutrients.

The sentence generated for expressing the overall evaluation is a single declarative sentence. In order to give a little bit of variation flavor in the syntactic shapes of the messages, we decided to

[1] We plan to add a suggestion on the future dishes to eat to the final message in next work.

use a negative copula for I_1, a declarative for I_2, and a positive copula for C_1 or C_2 and C_3. In particular, the overall evaluation is *non buono* (*not good)* or *non va bene* (*not OK*) when there is at least one macronutrient classified as I_1 or I_2, respectively. In alternative, the global evaluation is *buono* (*good*) or *molto buono* (*very good*) when there is at least one macronutrient classified as C_1 or C_2 respectively. Finally, the global evaluation is a *ottima scelta* (*great choice*) (see Table 1).

The sentence generated for expressing the appropriateness of the specific macronutrient follows a fixed schema too. It is a positive copula sentence with a predicate expressing the deviation *ricco/povero/perfetto* (*rich/poor/perfect*), and a PP modifier specifying the macronutrient, e.g. *in lipidi* (*in lipids*). Moreover, an adverb, e.g. *leggermente* (*lightly*) distinguishes C_2 and C_2 cases (see Table 1).

Note that both the overall and the specific macronutrient messages do not use referring expressions. Indeed, at this stage of the project we did not yet account for this specific feature.

Given the persuasive intent of the system, a crucial point concerns the persuasiveness of the messages by considering psychological theories. Many works in literature considered the application of NLG for presenting the results of automated reasoning to the user, e.g., (Weiner, 1980; Barzilay et al., 1998; Lacave and Diez, 2004). Moreover, many theories on the design of persuasive textual (and multimedia) messages have been proposed in the last years. We can split these studies in two narrow categories. The first category includes the theories approaching the persuasion from an empirical point of view, by using strategies and methods typical of the psychology and of the interaction design (Fogg, 2002; Reiter et al., 2003; Cialdini, 2009; Kaptein et al., 2012). The second category includes the theories approaching the persuasion from a theoretical point of view, by using strategies and methods typical of cognitive science (Hovy, 1988; de Rosis and Grasso, 2000; Guerini et al., 2007). Similar to (Kaptein et al., 2012), the Cialdini's general theory of persuasion has inspired our design of the messages (Cialdini, 2009). 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 1 belong to the patterns of authority and consistency. With respect to the low-level linguistic strategies, by following (de Rosis and Grasso, 2000), we used a number of adverbs, e.g. *davvero, molto, leggermente* (*really, very, lightly*) in order to enhance or mitigate a message. Furthermore, compared to Guerini et al. persuasive strategies taxonomy (Guerini et al., 2007), 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.

The sentences have been eventually realized by using the SimpleNLG-IT engine realizer, a porting of SimpleNLG for Italian language (Gatt and Reiter, 2009; Mazzei et al., 2016). So, the messages previously described have been primary encoded in the form of quasi-trees and secondary, after aggregation (Section 4.1) and word-lexicalization (Section 4.2), realized by using SimpleNLG-IT. There are several advantages to use SimpleNLG with respect to string templates in this specific project. The three majors advantages are: (i) we have a multilingual Italian/English version of the realiser, that allows to change language by simply switching from Italian to English lexicons, (ii) the design and implementation of the aggregation strategies are simpler, (iii) the diffusion of the Java language allows to integrate the generator into larger Java-based software platform.

In the next sections we describe the procedures of aggregation and lexicalization implemented by using the facilities exposed by SimpleNLG.

4.1 Aggregation strategies

The aggregation plays an important role to generate fluent and efficient texts (Reiter and Dale, 2000; Gatt and Krahmer, 2018). Moreover, in several domains, as healthcare or education, it has been proven that aggregation of the sentences improves the efficacy of the messages (McKeown et al., 1997; DiEugenio et al., 2005).

In the specific case of the MADiMan messages, aggregation can be performed in many ways since the messages concerning overall evaluation and

Category	Prototypical Message	English translation
I_1^+/I_1^-	Questo menù non è buono. Il menù è troppo *ricco/povero* in proteine.	This menu is not good. The menu is really *rich/poor* in proteins.
I_2^+/I_2^-	Questo menù non va bene. Il menù è *ricco/povero* in proteine.	This menu is not OK. The menu is *rich/poor* in proteins.
C_1^+/C_1^-	Questo menù è buono. Il menù è *ricco/povero* in proteine.	This menu is good. The menu is *rich/poor* in proteins.
C_2^+/C_2^-	Questo menù è molto buono. Il menù è leggermente *ricco/povero* in proteine.	This menu is very good. The menu is lightly *rich/poor* in proteins.
C_3	Questo menù è un'ottima scelta. Il menù è perfetto in proteine.	This menu is a great choice. The menu is perfect in proteins.

Table 1: The prototypical messages describing the STP reasoner classification for the caloric value for the proteins. The italicized text vary among $+/-$ deviation. The underlined text varies among the three macronutrients.

macronutrients often have very similar quasi-trees.

We write (O_C, O_L, O_P) to indicate the symbolic output for carbohydrates, lipids and proteins respectively, where $O_X \in \{I_1^-, I_1^+, I_2^-, I_2^+, C_1^-, C_1^+, C_2^-, C_2^+, C_3\}$. Indeed, a trivial aggregation strategy based on aggregation at the sentence level could merge only messages that belong to the same category, i.e. $O_x = O_y$: this trivial strategy corresponds to the *syntactic aggregation* in the classification of (Reape and Mellish, 1998). However, we design an aggregation strategy that accounts for a more sophisticated form of *conceptual aggregation*. The aggregation algorithm can be split in two parts, a *selection* and a *merging*.

Selection

In order to concentrate the focus on the most important information for the diet, the general idea of the selection is to give emphasis on the messages concerning incompatibility. So, during the selection step, if there are messages describing the incompatible value of a macronutrient, all the messages describing the compatible values will be removed. So, in the selection step there are three alternative cases:

A. There is a case of permanent inconsistence on one or more macronutrient: $\exists X \in \{C, L, P\} : O_X = I_1$

B. There is a case of occasional inconsistence on one or more macronutrient: $\forall X \in \{C, L, P\} : O_X \neq I_1 \wedge \exists Y \in \{C, L, P\} : O_Y = I_2$

C. All the three categories of macronutrients are consistent: $\forall X \in \{C, L, P\}\ \exists i \in \{1, 2, 3\} : O_X = C_i$

In the cases **A.** and **B.**, we aggregate the messages by exploiting the information about incompatibility, that is by removing the messages concerning the compatible macronutrients and by merging the messages about incompatible macronutrients. So, the final document will have one single overall sentence describing the inconsistence, and one merged message concerning the values of the inconsistent macronutrients. In the case **C.**, the final document will have one single overall sentence describing the minimal consistent value, and one merged message concerning the values of all the three macronutrients.

Merging

By taking into account the persuasive goals of the system, we decided to implement and test two different strategies to merge the specific messages concerning the macronutrients. In general there are many possible mechanisms to merge two sentences, i.e., simple conjunction, conjunction via shared participants, conjunction via shared structure, and syntactic embedding (Reiter and Dale, 2000). At this stage of the project, the system allows to use all these mechanisms but syntactic embedding. In particular, we decided to experimentally compare (see Section 5) the conjunction via shared structure on the VP constituent (VP-aggregation) and on the NP contained into prepositional phrase (set-aggregation). In other words, by considering the sentences (i) *The menu is perfect in proteins* and (ii) *The menu is perfect in lipids*, the VP-aggregation produces the sentence *The menu is perfect in proteins and is perfect in lipids* while the set-aggregation produces *The menu is perfect in proteins and lipids*.

We decided to use VP-aggregation and set-aggregation mechanisms since they have two spe-

cific features that could influence the persuasiveness of the final message. The VP-aggregation, by repeating the semantic predicate contained in the copula construction, could communicate in a more efficient way the (in)compatibility of a specific macronutrient. In contrast, the set-aggregation produces shorter messages that could be perceived as more natural and so more trustable. Note that VP-aggregation can be always applied independently by the compatibility values and the deviations expressed by the specific macronutrient messages. In contrast, we can apply set-aggregation only when the sentences have exactly the same syntactic shape, which corresponds to having the same value in compatibility and in deviation.

In Section 5 we will evaluate the appealing of messages built with two different aggregation strategies where the first (*all-VP* henceforth) always uses VP-aggregation and the second (*set+VP* henceforth) maximally uses set-aggregation in combination, in same cases, with VP-aggregation. In particular, in order to manage all the possible combinations of compatibility and deviations, for the *set+VP* strategy we follow this simple two-step algorithm:

1. Set-aggregate all the shape-equivalent sentences
2. VP-aggregate the sentence resulting from the first step (if any) with the remaining sentences.

For instance, the sentences *The menu is lightly rich in carbohydrates*, *The menu is rich in lipids*, *The menu is lightly rich in proteins*, will be aggregated in the *all-VP* strategy as *The menu is lightly rich in carbohydrates, is rich in lipids and is lightly rich in proteins*. In contrast, the same sentences will be aggregated in the *set+VP* strategy as *The menu is lightly rich in carbohydrates and proteins and is rich in lipids.*

Finally, note that in some cases we have a certain degree of freedom in the ordering of the aggregated sentences. We followed the idea to start with the most positive feedback, as suggested by some theories of persuasion (Steelman and Rutkowski, 2004; Dohrenwend, 2002). So, we decided to order the aggregated messages by considering their compatibility value. For instance, the sentences *The menu is poor in carbohydrates*, *The menu is lightly rich in lipids*, *The menu is lightly rich in proteins*, will be aggregated as *The menu is lightly rich in lipids and proteins and is poor in carbohydrates.*

4.2 Choosing words

Another feature that we implemented in realization is a trivial treatment of lexical variations. Indeed, many studies showed the importance and the complexity of the lexicalization task, e.g. (Stede, 1994; Reiter et al., 2005). In particular, an acceptable lexicalization procedure should take into account the contextual and stylistic constraints arising from all the possible words combinations (Gatt and Krahmer, 2018).

We think that variability could play an important role in the persuasive goal of the system. Since a constant lexical choice could be perceived as boring or artificial, for open-class categories (that are nouns, verbs, adjectives and adverbs) we decided to implement two different versions of the lexicalization procedure. The first lexicalization procedure that always associated one single word for each concept, and an alternative second lexicalization procedure that randomly associated one word choosing from a set of three possible words. In particular, for the Italian version of the realizer, the synonymous set has been decided by searching in the default Italian lexicon, that is a *simple* lexicon, i.e., a lexicon studied to be perfectly understood by most Italian people (Mazzei, 2016). We are aware that this trivial lexicalization procedure could give a sort of *cognitive dissonance* in some cases, but we believe that it could also improve the trustability of the system.

Also if the main focus of the experimental part of the paper concerns the experimental evaluation of the aggregation strategies, in Section 5 we provide also some user feedback about lexicon variability.

5 Experimental setting: the CheckYourMeal! app

We describe a first human-based experimentation produced with a small group of 20 users. The main goal of this experimentation was to give a realistic feedback about the appealing and, in some form, the persuasion strength of the message aggregation strategies. So, we designed a *game of diet* simulation (see below). We are aware that a scientific evaluation about the real efficacy of the persuasion power of the NLG should follow the scientific standards of the medical research field

(cf. (Reiter et al., 2003)). However, as pointed out by some research in the human computer interaction field, also pilot studies can give important feedbacks "especially when in the early stages of design or when evaluating novel technologies" (Klasnja et al., 2011; Hekler et al., 2013).

In order to create a realistic experimentation we designed and realized an app for mobile called *CheckYourMeal!* (Figure 3). In the current stage, *CheckYourMeal!* is still under development and it is used only for research purposes. So, it is not yet available as a commercial app.

CheckYourMeal! provides many standard functionalities of the *quantified self* domain app, as registration of username/password, log-in, insertion of personal and anthropometric data (e.g., age, weight, physical activity, etc.). The principal goal of the application is to help users in the management of their diets. The diet is considered as a number of constraints over the week (cf. Section 3). The week is scheduled as 21 slots to fill, i.e., breakfast, lunch and supper for each day from Monday to Sunday. For each slot of the week, a number of possible menus are presented to the user, and she can decide to eat one of them. The feedback about the compatibility of a specific menu is provided both in graphical and textual forms. The graphical feedbacks are (i) a cake-shaped diagram showing the caloric contents in carbohydrates, lipids and proteins, and (ii) three histograms showing their ideal values for that specific slot of the week. The textual feedbacks are two sentences automatically generated containing the overall evaluation and macronutrients evaluation respectively. In Figure 3 we report a screenshot of the app with the graphical (lower side) and textual (upper side) feedbacks. The experimentation was performed only in Italian.

We asked the users to interact with *CheckYourMeal!* by considering a simulation context. A user should imagine to eat for a period into a restaurant: for each slot of the week she has to choose only among the menus proposed in the app. In the simulation, the menus were randomly generated by considering the recipes of the Gedeone database, that is a collection of 500 Mediterranean recipes annotated with their caloric contents (Anselma et al., 2018).

Figure 3: A screenshot of the a message showed by CheckYourMeal! app.

Experimental protocol

We prepared an instruction sheet describing the game and the main goals of the experimentation. In particular, we explicitly informed the users that we wanted to compare two different versions of the NL message generator, the blue version and the violet version, but without any other information about the specific qualities that we wanted to test. The blue version corresponds to the *all-VP* aggregation strategy while the violet version corresponds to the *set+VP* aggregation strategy. We believe that with this briefing the testers could give more attention on the linguistic details of the textual feedback. Apart from the blue/violet version tests, we asked the testers to try also a feature called *variable lexicon* (see Section 4.2). We explicitly informed the testers that this feature was not our main experimental goal.

We asked the testers to play the diet game for a simulated period of two weeks, spending at least 15 minutes of their time. Moreover, we asked testers to play one week with the blue version and one week with the violet version. At the end of the experimentation, we asked the testers to compile a feedback form. The form was composed by 24 questions: 8 were multiple choices questions con-

cerning personal data; 4 were Likert-scale questions concerning the app and the lexicon; 9 were Likert-scale questions concerning the blue and violet versions of the generator; finally, 3 were open general questions concerning suggestions for possible improvements of the app, the feeling perceived and the lexicon.

The main hypothesis that we tested was about the appealing of the violet version with respect to the blue version. In particular in the form we wanted to compare four specific properties of the messages, that are *Usefulness, Persuasiveness Boringness, Easiness*. These specific four questions are[2]:

QU: Usefulness perceived: *The text messages in the blue version are more useful than the text messages in the violet version in order to make the best choice.*

QP: Persuasiveness perceived: *The text messages in the blue version are more persuasive than the text messages in the violet version.*

QB: Boringness perceived: *The text messages in the blue version are more boring than the text messages in the violet version.*

QE: Easiness perceived: *The text messages in the blue version are easier to understand than the text messages in the violet version.*

We used a Likert scale from 1 to 5 where 1=*I totally disagree* and 5=*I totally agree*.

In order to evaluate the feasibility of the experimental setting, we first tried the game with a preliminary group of three people: this pre-test suggested us to prepare a more detailed instruction sheet. Successively, we conducted the main study with a group of 20 people. All of them were Italian mother tongue, have provided their real anthropometric data, and have completed the test in a silent ambient after reading the instructions. Most of them were students or faculties in computer science and used a smartphone provided by us. We are aware that the small size and the homogeneity of the test group in this study does not allow to discover possible correlations between subgroup features (e.g. demographics) and final results.

[2]Translated form the original Italian questions.

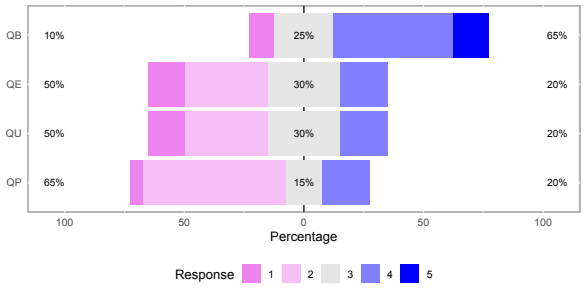

Figure 4: A plot showing the distribution of the answers to the questions QB, QE, QP, QU.

Results

In Figure 4 we report the distribution of the answers to the four questions which are the main goal of our experimentation[3].

The picture shows a quite clear preference for the violet version, which applies the *set+VP* aggregation strategy, with respect to the blue version, which applies the *all+VP* aggregation strategy. In other words, for all four properties, that are boringness (mean=3.60, s.d.=1.10), usefulness (mean=2.55, s.d.=1.00), persuasiveness (mean=2.50, s.d.=0.89), easiness (mean=2.55, s.d.=1.00), the shorter messages produced with the *set+VP* aggregation strategy are preferred with respect to the longer messages produced with the *all-VP* aggregation strategy. Indeed, we tested the statistical significance of the preference for the violet version with respect to the blue one (i.e., the answer has a numeric value < 3 for questions QE, QU, QP and > 3 for question QB). We obtained the (two-tailed) p-values 0.03, 0.03, 0.01, 0.01 for QE, QU, QP, QB respectively.

As post-hoc hypothesis we decided to analyze the result of the Likert-scale question concerning lexicon variability that is: *The "variable lexicon" option makes the use of the app more enjoyable.* (QV, 1=*I totally disagree* and 5=*I totally agree*.). In Figure 5 we report the distribution of the answers for QV (mean=3.40, s.d.=1.0). Also, if the distribution of the answers seems to show a preference for random lexical variations (the p-value for > 3 is 0.04), a specific experimentation is necessary to confirm this result.

An exploratory analysis of the responses given by the users gives us a feedback on the appealing of the app as a whole. In particular, we can infer

[3]The statistical analysis was performed by using the Likert package of R. We considered the points in the Likert scale as equidistant.

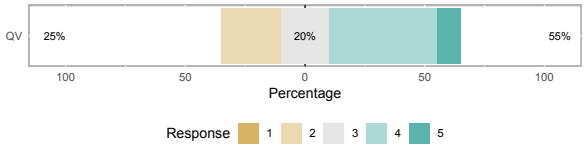

Figure 5: A plot showing the distribution of the answers for the questions QV.

from the distribution of the answers that (1) the user interface of the app is clear, (2) both graphics and text messages are perceived as useful to perform the best choice.

Finally, by reading the free comments section of the forms, an interesting speculation is that the two aggregation strategies have an appeal depending on the polarity of the messages. Indeed, some comments pointed out that the repetition of the predicate (*all-VP* strategy) gives a judgmental or blaming attitude to the virtual dietitian.

6 Conclusions and future work

In this paper we have presented a first human-based evaluation of a NL generator of persuasive messages in the diet management context. We have briefly described the main components of the MADiMan system and we have detailed the design and implementation system of the NLG module. Finally, we have described the details of a game-based simulation of the system by using the *CheckYourMeal!* app. By considering a number of perceived properties, the experimental results show preferences towards short messages obtained with a complex aggregation strategy.

In future work, we intend to perform the experimentation on a greater number of testers. In particular, in order to have more qualified feedback, we intend to evaluate the system with a group of undergraduate students in dietetics. Moreover, with more users we will be able to test several versions of the message generators, considering the variability of the lexicon too.

Another research question that we intend to follow regards the *explainability* of the answer. For tackling such issue, we intend to exploit the information regarding the past meals that the user has eaten during the week.

References

Luca Anselma and Alessandro Mazzei. 2015. 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*, pages 79–90.

Luca Anselma, Alessandro Mazzei, and Franco De Michieli. 2017. An artificial intelligence framework for compensating transgressions and its application to diet management. *Journal of Biomedical Informatics*, 68:58–70.

Luca Anselma, Alessandro Mazzei, and Andrea Pirone. 2018. Automatic reasoning evaluation in diet management based on an italian cookbook. In *Proceedings of the Joint Workshop on Multimedia for Cooking and Eating Activities and Multimedia Assisted Dietary Management*, CEA/MADiMa '18, pages 59–62, New York, NY, USA. ACM.

Regina Barzilay, Daryl Mccullough, Owen Rambow, Jonathan Decristofaro, Tanya Korelsky, Benoit Lavoie, and Cogentex Inc. 1998. A new approach to expert system explanations. In *9th International Workshop on Natural Language Generation*, pages 78–87.

Daniel Braun, Ehud Reiter, and Advaith Siddharthan. 2015. 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*, pages 156–165.

Daniel Braun, Ehud Reiter, and Advaith Siddharthan. 2018. Saferdrive: An nlg-based behaviour change support system for drivers. *Natural Language Engineering*, 24(4):551–588.

Robert B. Cialdini. 2009. *Influence: science and practice*. Pearson Education.

Patricia Conde-Clemente, Jose M. Alonso, and Gracian Trivino. 2018. Toward automatic generation of linguistic advice for saving energy at home. *Soft Computing*, 22(2):345–359.

Rina Dechter, Itay Meiri, and Judea Pearl. 1991. Temporal constraint networks. *Artif. Intell.*, 49(1-3):61–95.

Barbara DiEugenio, Davide Fossati, Dan Yu, Susan M. Haller, and Michael Glass. 2005. 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*, pages 50–57.

Anne Dohrenwend. 2002. Serving up the feedback sandwich. *Family practice management*, 9(10):43–50.

B.J. Fogg. 2002. *Persuasive Technology. Using computers to change what we think and do*. Morgan Kaufmann Publishers, Elsevier.

Albert Gatt and Emiel Krahmer. 2018. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *J. Artif. Intell. Res.*, 61:65–170.

Albert Gatt and Ehud Reiter. 2009. SimpleNLG: A Realisation Engine for Practical Applications. In *Proceedings of the 12th European Workshop on Natural Language Generation*, ENLG '09, pages 90–93, Stroudsburg, PA, USA. Association for Computational Linguistics.

Marco Guerini, Oliviero Stock, and Massimo Zancanaro. 2007. A taxonomy of strategies for multimodal persuasive message generation. *Applied Artificial Intelligence*, 21(2):99–136.

Eric B. Hekler, Predrag Klasnja, Jon E. Froehlich, and Matthew P. Buman. 2013. Mind the theoretical gap: Interpreting, using, and developing behavioral theory in hci research. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 3307–3316, New York, NY, USA. ACM.

Eduard H. Hovy. 1988. *Generating Natural Language Under Pragmatic Constraints*. Lawrence Erlbaum, Hillsdale, NJ.

Maurits Kaptein, Boris E. R. de Ruyter, Panos Markopoulos, and Emile H. L. Aarts. 2012. Adaptive persuasive systems: A study of tailored persuasive text messages to reduce snacking. *TiiS*, 2(2):10.

Predrag Klasnja, Sunny Consolvo, and Wanda Pratt. 2011. How to evaluate technologies for health behavior change in hci research. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 3063–3072, New York, NY, USA. ACM.

Carmen Lacave and Francisco J. Diez. 2004. A review of explanation methods for heuristic expert systems. *Knowl. Eng. Rev.*, 19(2):133–146.

LARN. 2014. *LARN - Livelli di Assunzione di Riferimento di Nutrienti ed energia per la popolazione italiana - IV Revisione*. SICS Editore, Italy.

Alessandro Mazzei. 2016. Building a computational lexicon by using SQL. In *Proceedings of Third Italian Conference on Computational Linguistics (CLiC-it 2016) & Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016), Napoli, Italy, December 5-7, 2016.*, volume 1749, pages 1–5. CEUR-WS.org.

Alessandro Mazzei, Cristina Battaglino, and Cristina Bosco. 2016. SimpleNLG-IT: adapting SimpleNLG to Italian. In *Proceedings of the 9th International Natural Language Generation conference*, pages 184–192, Edinburgh, UK. Association for Computational Linguistics.

Kathleen R. McKeown, Shimei Pan, James Shaw, Desmond A. Jordan, and Barry A. Allen. 1997. Language generation for multimedia healthcare briefings. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, ANLC '97, pages 277–282, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mike Reape and Chris Mellish. 1998. Just what is aggregation anyway? In *ENLG 1998 - Proceedings of the European Workshop on Natural Language Generation*.

E. Reiter, R. Robertson, and L. Osman. 2003. Lessons from a Failure: Generating Tailored Smoking Cessation Letters. *Artificial Intelligence*, 144:41–58.

Ehud Reiter. 2007. An architecture for data-to-text systems. In *Proc. of the 11th European Workshop on Natural Language Generation*, ENLG '07, pages 97–104, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press, New York, NY, USA.

Ehud Reiter, Somayajulu Sripada, Jim Hunter, and Ian Davy. 2005. Choosing words in computer-generated weather forecasts. *Artificial Intelligence*, 167:137–169.

Fiorella de Rosis and Floriana Grasso. 2000. Affective interactions. chapter Affective Natural Language Generation, pages 204–218. Springer-Verlag, New York, NY, USA.

Manfred Stede. 1994. Lexicalization in natural language generation: A survey. *Artificial Intelligence Review*, 8(4):309–336.

Lisa A. Steelman and Kelly A. Rutkowski. 2004. Moderators of employee reactions to negative feedback. *Journal of Managerial Psychology*, 19(1):6–18.

J. L. Weiner. 1980. Blah, A system which explains its reasoning. *Artif. Intell.*, 15(1-2):19–48.