

CheckYourMeal!: ragionamento automatico e generazione automatica del linguaggio per la gestione della dieta alimentare

Luca Anselma, Alessandro Mazzei

Dip. di Informatica, Università degli studi di Torino

anselma@di.unito.it, mazzei@di.unito.it

Abstract

CheckYourMeal! è un'app che fa parte di un sistema di intelligenza artificiale per assistere gli utenti nella loro interazione con il cibo. CheckYourMeal! permette di mostrare i risultati del ragionamento automatico sia in forma grafica che testuale.

1 Introduzione

Seguire una dieta sana è un punto chiave per avere una vita sana. L'intelligenza artificiale e il pervasive computing sono due tecnologie emergenti che possono aiutare le persone a mangiare in modo corretto. Inoltre, le tecniche di generazione del linguaggio naturale (NLG) possono essere utilizzate nel contesto della dieta per spiegare i risultati di ragionatori numerici e simbolici (ad es. [2]) o per motivare gli utenti a effettuare la migliore scelta alimentare.

MADiMan (Multimedia Application for Diet Management, <http://di.unito.it/madiman>) è un progetto nato l'obiettivo di costruire un dietista virtuale in grado di recuperare le informazioni nutrizionali direttamente da una specifica ricetta, ragionare su ricette e diete con flessibilità, permettendo alcune forme di disobbedienza dalla dieta, e infine persuadere l'utente a minimizzare tali atti di disobbedienza [3]. L'architettura di MADiMan è composta da vari moduli: un'app, un modulo di estrazione delle informazioni utilizzato per ricavare i valori dei nutrienti di una specifica ricetta, un ragionatore numerico che verifica la compatibilità di un piatto con la dieta e rimodula dinamicamente la dieta stessa per i pasti futuri [2], e un servizio NLG che converte i risultati del calcolo in forma testuale [1].

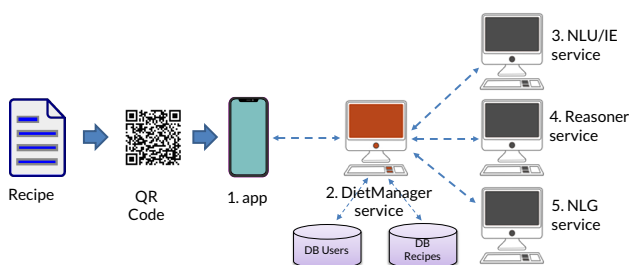


Figura 1: The MADiMan architecture.

2 Ragionamento STP su dieta e cibo

Le raccomandazioni dietetiche sono spesso espresse come un sistema di intervalli di riferimento per i nutrienti da seguire per un tempo significativo. Ad esempio, il fabbisogno energetico complessivo per un uomo di 22 anni che pesa 73 kg ed è alto 1,84 m e pratica un'attività fisica moderata è 2450 kcal/g [5]. La quantità raccomandata di macronutrienti [5] è di 263 kcal/g di proteine, 735 kcal/g di lipidi e 1452 kcal/g di carboidrati. In [2] abbiamo studiato come rappresentare tali valori dietetici in un framework di intelligenza artificiale chiamato STP [4]. Gli STP si basano su vincoli di tipo $c \leq x - y \leq d$, dove x e y sono variabili e c e d sono numeri il cui dominio può essere discreto o continuo. Un vincolo STP può essere interpretato come "la distanza tra i punti temporali x e y deve essere compresa tra c e d ". La consistenza di un STP può essere verificata in modo efficiente con un algoritmo di calcolo dei percorsi minimi tra tutte le coppie come quello di Floyd-Warshall [4].

Nella nostra formalizzazione modelliamo un pasto come la quantità di macronutrienti assunta tra un punto temporale m_S (l'inizio del pasto) e un punto temporale m_E (fine pasto). Analogamente, utilizziamo i punti temporali day_{i_S} e day_{i_E} e $week_S$ e $week_E$ per rappresentare rispettivamente i pasti in un giorno day_i e nella settimana. Grazie a STP possiamo consentire agli utenti di effettuare trasgressioni alla dieta e offrire supporto per compensare in seguito tali trasgressioni. Infatti STP consente di imporre vincoli meno stringenti nei periodi più brevi (ad es. giorni o pasti) e vincoli più rigidi per i periodi più lunghi (ad es. settimane) in modo da permettere deviazioni nel breve termine continuando comunque a perseguire gli obiettivi a lungo termine della dieta. Ogni volta che viene consumato un pasto, l'STP viene aggiornato e il meccanismo di propagazione dei vincoli aggiornerà anche i vincoli STP relativi ai pasti futuri [2]. Ad esempio, la raccomandazione di 1452 kcal/g di carboidrati determina i vincoli STP (considerando l'intervallo di una settimana e consentendo una tolleranza dai valori di σ_w , σ_d e σ_m rispettivamente per la settimana, il giorno e il pasto, e rappresentando con un pedice S o E l'inizio e la fine di una settimana/giorno/pasto):

-per la settimana: $1452 \cdot 7 \cdot (1 - \sigma_w) \leq week_E - week_S \leq 1452 \cdot 7 \cdot (1 + \sigma_w)$

-per ogni giorno della settimana day_i , $i = 1, \dots, 7$, $1452 \cdot (1 - \sigma_d) \leq day_{i_E} - day_{i_S} \leq 1452 \cdot (1 + \sigma_d)$,



Figura 2: Schermate di CheckYourMeal.

-per ogni pasto m_{ij} del giorno day_i , considerando che il pasto dovrebbe fornire una quota calorica $quota_{ij}$, $1452 \cdot quota_{ij} \cdot (1 - \sigma_m) \leq m_{ijE} - m_{ijS} \leq 1452 \cdot quota_{ij} \cdot (1 + \sigma_m)$.

È importante notare che medici e dietisti possono non solo variare i parametri σ_w , σ_d , σ_m e $quota_{ij}$, ma sono anche liberi di fornire vincoli differenti per pasti o giorni specifici e di personalizzare i requisiti energetici totali e la quantità dei macronutrienti.

Anche il cibo che l'utente effettivamente assume può essere rappresentato in modo omogeneo nel framework STP semplicemente rappresentando i valori nutrizionali di un pasto come vincoli STP. Questo permette di offrire supporto a casi in cui le porzioni o la composizione del pasto non sono noti con precisione. Determinare la consistenza dell'STP corrisponde a verificare se un pasto soddisfa i vincoli del pasto, del giorno e della settimana, e la rete minima risultante [4] fornisce i nuovi vincoli impliciti per i pasti futuri che consentono comunque di raggiungere l'obiettivo settimanale [2].

3 CheckYourMeal! e il modulo di NLG

CheckYourMeal! è un'app per iOS, sviluppata nel linguaggio funzionale Swift, progettata e realizzata per presentare ad un utente l'output di un sistema di ragionamento automatico che lavora sul cibo assunto e sulla dieta che l'utente intende seguire. CheckYourMeal! consente una presentazione dei risultati sia grafica (diagrammi) che testuale (messaggi).

In Fig. 2 riportiamo tre screenshot dell'app. L'interfaccia utente è strutturata in tre sezioni: (1) la sezione Home, in cui agli utenti sono fornite informazioni generali, (2) la sezione Menu, dove gli utenti possono vedere i suggerimenti per il pasto, e dove possono inserire i pasti scelti, e (3) la sezione Profilo, dove gli utenti possono modificare le impostazioni e i parametri dell'utente. Nella sezione Menu (schermata centrale in Fig. 2), all'utente vengono presentati alcuni suggerimenti sull'adeguatezza di una serie di menù. I menù vengono costruiti partendo da un database di ricette appartenenti alla tradizione mediterranea, e vengono ordinati nella presentazione in base alla distanza, in termini di calorie dei macronutrienti, dai valori che la dieta prevede come ideali per quello specifico pasto. Quando un utente seleziona un menu specifico, CheckYourMeal! mostra sia (1) un grafico a torta, sia (2) un messaggio testuale che contiene le informazioni sui valo-

ri dei macronutrienti del menu scelto (schermata destra nella Fig. 2).

Il server (*DietManager service* in Fig. 1) è stato scritto in Java e utilizza il framework Spring per comunicare con la app CheckYourMeal!. Il server chiama il modulo NLG come una libreria Java esterna integrata in un singolo file jar. Il modulo di NLG è composto da due sottomoduli: (i) un sistema a regole che implementa contemporaneamente sia il *document planner* sia il *sentence planner* [3] e, (ii) un *realizer* bilingue inglese/italiano basato sulla libreria SimpleNLG-it. L'intero servizio NLG è stato sviluppato utilizzando il linguaggio clojure, ovvero un linguaggio funzionale che viene eseguito sulla JVM.

Sia il document planner che il sentence planner seguono uno schema generale per i messaggi in output. Infatti, tutti i messaggi saranno composti da due parti: una valutazione complessiva del piatto, e tre valutazioni specifiche sui carboidrati, lipidi, proteine. Le frasi generate per esprimere l'appropriatezza dei macronutrienti specifici sono frasi di copula positive con un predicato che esprime la deviazione (cioè ricco/povero/perfetto) e un modificatore preposizionale che specifica il macronutriente (ad esempio *nei lipidi*). Inoltre, un avverbio (ad esempio *leggermente*) distingue le deviazioni dalla scelta ottimale. Si noti che i sentence plan generati sono, a parte il lessico, indipendenti dalla lingua usata nei messaggi. Quindi, la selezione dell'utente della lingua per i messaggi (inglese o italiano) corrisponde semplicemente all'utilizzo di una classe di realizzatore diversa di SimpleNLG-it. L'effettiva implementazione del generatore consente inoltre di selezionare altre due caratteristiche riguardanti il lessico (fisso o variabile) e la strategia di aggregazione (VP e SET). Abbiamo testato le varie funzionalità di CheckYourMeal!, in un primo esperimento di valutazione umana con 20 persone [1]. In futuro, intendiamo anche integrare informazione ontologica nel sistema di ragionamento.

Riferimenti bibliografici

- [1] Luca Anselma and Alessandro Mazzei. Designing and testing the messages produced by a virtual dietitian. In *Proc. of 11th International Conference on Natural Language Generation (INLG 2018)*, pages 244–253. ACL, 2018.
- [2] Luca Anselma, Alessandro Mazzei, and Franco De Micheli. An artificial intelligence framework for compensating transgressions and its application to diet management. *Journal of Biomedical Informatics*, 68:58–70, April 2017.
- [3] Luca Anselma, Alessandro Mazzei, and Andrea Pirone. Automatic reasoning evaluation in diet management based on an italian cookbook. In *Proc. of the IJCAI Joint Workshop CEA/MADiMa '18*, pages 59–62. ACM, 2018.
- [4] Rina Dechter, Itay Meiri, and Judea Pearl. Temporal constraint networks. *Artif. Intell.*, 49(1-3):61–95, May 1991.
- [5] LARN. *LARN - Livelli di Assunzione di Riferimento di Nutrienti ed energia per la popolazione italiana - IV Revisione*. SICS Editore, Italy, 2014.