

Compteur de calories

Résumés des articles

Aurélien van Delft
Raphaël Vander Marcken
Jérôme Hellinckx
Thomas Herman

10 novembre 2015

Table des matières

1	<i>Challenges for Nutrition Recommender Systems</i>	2
1.1	Propos	2
1.2	Pourquoi mêler systèmes de recommandations et diététique ?	2
1.3	Les différents types de systèmes de recommandations nutritionnelles	2
1.3.1	Recommandation par repas	2
1.3.2	Recommandation par aliment	2
1.4	Challenges pour les systèmes de recommandations nutritionnelles	2
1.4.1	Informations sur l'utilisateur	2
1.4.2	Base de données de recettes et leur valeur nutritionnelle	3
1.4.3	L'ensemble de contraintes et de règles	3
2	<i>Deriving a Recipe Similarity Measure for Recommending Healthful Meals</i>	3
2.1	Propos	3
2.2	Les différentes approches pour recommander des recettes	3
2.3	Représentation binaire d'une recette	4
2.4	La mesure de similarité	4

Article 1: *Challenges for Nutrition Recommender Systems*

1.1 Propos

L’auteur donne un bref aperçu des différentes manières d’implémenter un système de recommandation nutritionnelle avant de passer en revue les différents problèmes rencontrés lors de la construction d’un tel système. Des éléments de réponse à ces problèmes sont également fournis.

1.2 Pourquoi mêler systèmes de recommandations et diététique ?

Il existe aujourd’hui de nombreuses sources d’informations concernant la nutrition et la diététique (sites webs gouvernementaux, forums de discussion, magazines en ligne, base de donnée de recettes, etc), permettant à chacun d’entreprendre une diététique saine. L’utilisateur se retrouve cependant confronté à une quantité confuse et conflictuelle d’informations, et ne dispose pas de l’expertise nécessaire pour interpréter cette information, et ne sait en conséquence pas quoi faire ou comment changer sa diététique. Une solution potentielle est de développer des systèmes de recommandations, effectuant automatiquement des suggestions adéquates basées sur le profil de l’utilisateur.

1.3 Les différents types de systèmes de recommandations nutritionnelles

1.3.1 Recommandation par repas

Le premier type de système de recommandation nutritionnelle se focalise sur la suggestion de repas. Ce genre de système utilise soit des mesures de similarité pour recommander des recettes selon les préférences de l’utilisateur (en se basant sur les ingrédients ou les notes d’utilisateur) soit des besoins nutritionnels de l’utilisateur pour conseiller des recettes qui satisfont ces besoins.

1.3.2 Recommandation par aliment

L’autre type de système ignore complètement les préférences de l’utilisateur et propose des aliments plus sains pour remplacer les aliments consommés actuellement par l’utilisateur.

1.4 Challenges pour les systèmes de recommandations nutritionnelles

1.4.1 Informations sur l’utilisateur

Obtenir des informations sur l’utilisateur est essentiel au bon fonctionnement d’un système de recommandation nutritionnelle. Typiquement, les données nécessaires sont :

1. les besoins nutritionnels de l’utilisateur. Ces informations peuvent généralement être trouvées sur le site de l’Organisation mondiale de la Santé.
2. des notes de l’utilisateur pour des repas ou aliments spécifiques. Afin d’effectuer des recommandations adéquates, le système a besoin de connaître les préférences de l’utilisateur. Il est donc nécessaire de collecter ces notes, en veillant à ce que l’effort fourni par l’utilisateur soit minimum (irréaliste de noter chaque repas ou aliment). De plus, lorsque le système est utilisé pour la première fois par l’utilisateur, ce système ne dispose que d’un nombre limité d’informations sur cet utilisateur et a besoin de temps pour accumuler suffisamment de données pour améliorer les recommandations. Cependant, le système devrait être capable de fournir des recommandations satisfaisantes même durant cette progression initiale, et donc avec très peu d’informations. Ceci est appelé le problème du départ à froid (*cold-start problem*). Il existe deux manières de régler ce problème. La première est d’augmenter le nombre de notes utilisateur collectées avant le premier

emploi du système, augmentant donc aussi l'effort fourni par l'utilisateur. La deuxième est d'utiliser l'information concernant les repas précédents de l'utilisateur pour utiliser des mesures de similarité pour recommander de nouvelles recettes. Ceci augmente aussi l'effort de l'utilisateur car demande de fournir en continu de l'information sur les repas précédents. Or, un plus grand effort réduit le désir d'utiliser le système, qui à son tour mène à moins d'information sur l'utilisateur. L'utilisateur devrait donc pouvoir voir les bénéfices d'une utilisation continue du système, augmentant sa volonté d'investir autant de temps et d'effort nécessaire au système. Un moyen de parvenir à ce but est de fournir de l'information quant à la progression ainsi que de l'information personnalisée.

3. de l'information sur les derniers repas de l'utilisateur. La validité de cette information peut-être questionnable (sous-estimation de la consommation ou même oubli total). FoodLog est une solution potentielle à ce problème.

Une importante question se pose pour les deux derniers cas : comment persuader l'utilisateur de continuer à noter des repas et à renseigner le système sur les derniers plats consommés ?

1.4.2 Base de données de recettes et leur valeur nutritionnelle

Le système nécessite d'avoir accès à une base de donnée avec assez de recettes pour tenir compte des préférences des utilisateurs et pour pouvoir varier les repas proposés, tout en garantissant un temps de réponse pour la recommandation acceptable. Il faut également s'assurer que les données obtenues depuis les tables nutritionnelles soient correctes. Comparer différentes sources est donc essentiel pour vérifier l'exactitude des données.

1.4.3 L'ensemble de contraintes et de règles

Au plus de contraintes et de règles sont imposées à l'algorithme, au plus adéquates seront les recommandations. Cependant, un nombre trop importants de contraintes entraîne un temps de réponse plus grand, surtout avec une large base de donnée de recettes étant donné que chaque recette doit être vérifiée pour chaque contrainte. Des contraintes contradictoires doivent aussi faire l'objet d'une attention particulière puisqu'elles pourraient empêcher l'algorithme de trouver une solution. Le problème sous-jacent est donc d'équilibrer le système pour allier assurance de la qualité de la recommandation avec performance du système.

Article 2: *Deriving a Recipe Similarity Measure for Recommending Healthful Meals*

2.1 Propos

Cet article explique les différentes expériences menées pour développer une mesure de similarité centrée sur l'utilisateur entre des recettes. À noter qu'on établit un lien évident entre mesure de similarité et un algorithme de recommandation ayant une approche basée sur le contenu.

2.2 Les différentes approches pour recommander des recettes

Plusieurs méthodes ont été considérées pour recommander des recettes :

1. par *collaborative filtering*, basé sur des notes de recettes partagées.
2. *content-based*, qui se concentre sur le contenu même d'une recette. Différentes approches existent pour comparer des recettes :

- (a) les notes de l'utilisateur pour certaines recettes déterminent les notes pour les ingrédients individuels permettant au système de noter d'autres recettes selon leurs ingrédients.
- (b) il est possible de considérer aussi différents niveaux d'importance pour les ingrédients. Les ingrédients ayant une importance plus grande auront donc une plus large contribution au score de similarité.
- (c) une recette est modélisée par un graphe en utilisant des ingrédients et la préparation. Une recette est représentée par un graphe avec des objets (ingrédients) et actions (frir, rôtir, etc.). La similarité entre deux recettes est donc exprimée en comparant les graphes de ces recettes.

Dans cet article, l'auteur propose de revisiter la modélisation d'une recette en proposant une approche centrée sur l'utilisateur (*content-based* donc).

2.3 Représentation binaire d'une recette

Étant donné que le but est d'avoir une approche centrée utilisateur pour comparer des recettes, il faut déterminer les aspects les plus importants d'une recette qui, selon les utilisateurs, contribuent le plus à la similarité. Pour ce, une expérience est menée où des amateurs cuisiniers se soumettent à un tri par carte où il leur est demandé d'organiser des recettes par groupe et de caractériser ce groupe verbalement (oriental, temps de préparation, viande, etc).

Cette répartition permet de créer un ensemble de caractéristiques pouvant prendre certaines valeurs. Cet ensemble de caractéristique permet de modéliser une recette selon une suite de booléens indiquant la présence ou non de certaines valeurs. On remarque que dans cette liste de valeurs ne figurent pas tous les ingrédients que pourraient contenir une recette mais plutôt une série d'attributs importants pour l'utilisateur.

2.4 La mesure de similarité

L'expérience du tri par carte a permis de constater que chaque propriété du vecteur de booléens modélisant une recette n'intervient pas avec la même influence dans la similarité perçue entre différentes recettes. Un poids est donc assigné à chaque caractéristique. Ce poids est donné par

$$w_i = \frac{c_i \sum_{j=0}^n c_j}{n} \quad (1)$$

où w_i est le poids de la valeur i , c_i l'occurrence de la valeur i dans le tri par carte, c_j l'occurrence de la valeur j appartenant à la même caractéristique que i et n le nombre de valeurs pour la caractéristique de i . La similarité entre deux recettes est alors obtenues avec une distance euclidienne pondérée

$$Dist(x, y) = \sum_i w_i (x_i - y_i)^2 \quad (2)$$

avec x et y les recettes, w_i le poids de la valeur i , x_i et y_i des booléens indiquant la présence ou non de la valeur dans les recettes.