

UNIVERSITÉ LIBRE DE BRUXELLES  
Faculté des Sciences  
Département d'Informatique

# Compteur de calories Rapport première partie

Aurélien van Delft  
Raphaël Vander Marcken  
Thomas Herman  
Jérôme Hellinckx

Superviseurs : Fabrizio Carcillo, Antonio Colaprico, Tom Lenaerts

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Pertinence des articles</b>	<b>3</b>
2.1	<i>Collaborative Filtering Recommender Systems</i> [1] . . . . .	3
2.2	<i>The Long Tail</i> [2] . . . . .	3
2.3	<i>Content-based recommendation systems</i> [3] . . . . .	3
2.4	<i>Challenges for Nutrition Recommender Systems</i> [4] . . . . .	3
2.5	<i>Deriving a recipe similarity measure for recommending healthful meals</i> [5] . . . .	3
2.6	<i>Hybrid Web Recommender Systems</i> [6] . . . . .	4
2.7	<i>Allowing for never and episodic consumers when correcting for error in food record measurements of dietary intake</i> [7] . . . . .	4
<b>3</b>	<b>Description de l'implémentation</b>	<b>4</b>

# 1 Introduction

Un compteur de calories est une application qui va permettre à un utilisateur de surveiller et améliorer sa diététique. Le logiciel gardera un historique de consommation et d'activités physiques de l'utilisateur, qu'il devra lui-même fournir, afin d'ajuster les recommandations qui lui seront suggérées (à sa volonté). Lesdites données que l'utilisateur entrera dans l'application seront utilisées pour déterminer ses besoins nutritionnels (et donc caloriques). Le système adaptera également ses recommandations en fonction des préférences de l'utilisateur qui seront évaluées via les repas qu'il a déjà consommés.

Il ressort du paragraphe précédent que le principal axe de recherche lié à l'application porte sur l'étude et la mise en place d'un système de recommandation. Un système de recommandation définit la manière dont l'application va proposer différentes recettes aux utilisateurs en fonction de son type. On retrouve principalement trois grandes classes de systèmes de recommandation.

1. Les systèmes de type *collaborative filtering* qui vont prendre en considération l'environnement des utilisateurs (i.e. les préférences des autres utilisateurs).
2. Les systèmes dits *content-based* qui se basent sur les objets en eux-mêmes (ici, basés sur les aliments).
3. Et enfin ceux définis comme *knowledge-based*, qui utilisent comme critères des informations obtenues plus explicitement (en interagissant directement avec un utilisateur par exemple).

Des systèmes hybrides regroupant plusieurs type de systèmes peuvent également être mis en place. Ces différents systèmes seront définis plus en détails par la suite.

## 2 Pertinence des articles

### 2.1 *Collaborative Filtering Recommender Systems* [1]

Cet article présente et explique les systèmes de recommandation de type "collaborative filtering". Il s'agit d'une des catégories principales de système de recommandation. L'approche collaborative prend en considération l'environnement des utilisateurs. Il est donc primordiale d'implémenter un système au minimum partiellement collaboratif afin d'effectuer des propositions plus précises dès lors que l'on dispose d'un ensemble d'utilisateurs. Cet article nous aidera à choisir et à implémenter un système collaboratif.

### 2.2 *The Long Tail* [2]

Cet article nous parle du "Long Tail". Ce dernier est un effet que l'on observe lorsqu'on regarde les ventes d'un secteur donné (livres, CD's, etc.). L'idée est que certains produits (~20%) se vendent beaucoup plus que les ~80% restants, ces derniers forment le "Long Tail". Ce sont tous les produits non grand public mais qui trouveront toujours une demande. Dans le cadre de ce projet, on pourra considérer le "Long Tail" comme étant formé des aliments plus exotiques, ceux qui ne sont pas choisis par la grande majorité des utilisateurs. Afin d'être le plus efficace possibles, trois règles ont été définies : rendre le maximum de produits disponibles, proposer le plus petit prix possible et rendre les produits facile à trouver. Cette dernière règle est approfondie en expliquant que lors de la sélection des candidats pour la prédiction de recommandation, on fait en sorte de s'enfoncer de plus en plus dans le "Long Tail", de manière à amener progressivement l'utilisateur à essayer des produits (aliments, dans notre cas) de moins en moins grand public.

### 2.3 *Content-based recommendation systems* [3]

Cet article présente et explique les bases des systèmes de recommandation de type "content-based". L'approche content-based se base sur les objets en eux-mêmes et sur des profils utilisateurs. Afin d'effectuer une recommandation, le système va représenter un objet comme une série d'attributs et va ensuite établir un profil pour un utilisateur donné. Ce dernier se verra attribuer ledit objet si une similitude suffisante est trouvée entre le profil et l'objet. Cet article présente un état de l'art des différentes techniques de création de profils et de recommandation dans un système content-based. L'approche content-based est utilisable même si le nombre d'utilisateurs est restreint, ce qui n'est pas le cas de l'approche collaborative-filtering.

### 2.4 *Challenges for Nutrition Recommender Systems* [4]

Comme l'indique son titre, cet article met en lumière les différents problèmes auxquels peut se confronter un système de recommandation quand il est utilisé dans un cadre nutritionnel. Plus particulièrement, l'auteur fournit une liste d'informations utilisateurs requises pour établir un système de recommandation (besoins nutritionnels, notes de repas, etc). Par ailleurs, il est mention de balance entre base de donnée et contraintes d'une part et performance de l'application d'autre part. Cet article est en conclusion intéressant parce qu'il précise un certain nombre de tâches à effectuer lors de la création d'un système de recommandation nutritionnel efficace.

### 2.5 *Deriving a recipe similarity measure for recommending healthful meals* [5]

Une des approches des systèmes de recommandations est *content-based*, c'est-à-dire qu'on compare le contenu même des objets considérés. Dans le cadre de ce projet, on cherche à

recommander des repas, la question est donc : comment comparer des repas entre eux et leur donner des scores de similarité afin de proposer des recommandations content-based ? L'article propose justement un moyen de comparer des recettes, et met en lumière d'autres méthodes utilisées également. Ledit article permet donc d'avoir une bonne idée générale des méthodes de comparaison et de modélisation de recettes tout en fournissant la sienne. Pourquoi avoir donc choisi cet article en particulier ? Il s'avère en fait que l'idée expérimentée ici est intéressante car elle propose de modéliser les recettes selon une approche centrée utilisateur. Au lieu de comparer uniquement des ingrédients entre eux, cette méthode détermine une liste de facteurs pondérés mise au point par des cuisiniers amateurs, cette liste constituée notamment d'ingrédients, du temps de préparation, de la difficulté de la recette, etc. Les recommandations basées sur cette approche de comparaison pourraient donc s'avérer plus précises. Notons que ceci nous permettra de ne pas proposer que des aliments, mais plutôt de proposer un ensemble aliments-recettes à l'utilisateur, et cela afin de rendre le système plus abouti.

## 2.6 *Hybrid Web Recommender Systems* [6]

Cet article traite des hybridations au sein des systèmes de recommandation. En effet, les différentes techniques de recommandation telles que collaboratives, "knowledge-base" ou encore "content-based", peuvent être imbriquées afin de former un système hybride. Sept techniques différentes pour imbriquer les techniques entre-elles sont examinées comme par exemple la "Cascade" qui met les différentes techniques de recommandation utilisées en relation avec un système de priorités les unes par rapport aux autres. L'article nous montre également comment différentes hybridations sont comparées et évaluées, ce qui va nous permettre d'avoir une certaine base sur laquelle s'appuyer.

## 2.7 *Allowing for never and episodic consumers when correcting for error in food record measurements of dietary intake* [7]

L'application garde un historique des consommations et activités sportives de l'utilisateur. Le système se base sur ses données pour ensuite évaluer ses besoins nutritionnels. Il est évidemment irréaliste d'estimer que l'historique fourni par cet utilisateur sera complet et donc vide de "trous". C'est pourquoi il faut analyser et mettre en place une méthode permettant de remplir ces trous dans l'historique, ce que propose l'article. Ajoutons également que des erreurs d'appréciation de consommation (entrées par l'utilisateur) peuvent apparaître, ce qui est aussi anticipé par la méthode proposée ici.

# 3 Description de l'implémentation

Le projet ayant pour but principal l'étude de différents algorithmes de recommandation, nous allons, dans un premier temps, implémenter des systèmes de recommandation monotype afin de les comparer. Dans un second temps, nous nous intéresserons à différentes hybridations possibles, plus complexes, pour observer les différents avantages/inconvénients qui en découlent sur la qualité des recommandations.

Une base de donnée sera nécessaire afin de stocker des informations comme les profils utilisateurs, les données concernant les aliments, etc. C'est pourquoi un serveur sera nécessaire et formera le cœur de l'application. Le serveur permettra de manipuler les données et d'appliquer les algorithmes de recommandation. Il permettra également d'échanger des données avec les utilisateurs de l'application qui sera une interface spécifique à la plateforme cible.

Notre choix concernant le langage utilisé pour l'implémentation du serveur se dirige plutôt

vers le Java pour sa simplicité d'utilisation concernant les algorithmes et l'implémentation d'un serveur. Le serveur sera en interaction avec la base de donnée via SQL. Côté application mobile, notre choix se porte sur l'utilisation de *Phonegap* qui nous permet d'écrire l'application avec une combinaison de Javascript et HTML/CSS. PhoneGap nous permet ensuite de générer l'application à partir de ce code pour différentes plateformes mobiles telles que Android et iOS.

## Références

- [1] Michael D. Ekstrand, and John T. Riedl, and Joseph A. Konstan. *Collaborative Filtering Recommender Systems*, Foundations and Trends in Human-Computer Interaction 4 (2010) : 81-173.
- [2] Chris Anderson. *The Long Tail*, <http://www.wired.com/2004/10/tail/>, January 10, 2004.
- [3] M. J. Pazzani and D. Billsus. *Content-based recommendation systems*, The Adaptive Web (Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl, eds.), Lecture Notes in Computer Science, vol. 4321, Springer, 2007, pp. 325–341.
- [4] Stefanie Mika. *Challenges for Nutrition Recommender Systems*, Proceedings of the 2nd Workshop on Context Aware Intel. Assistance, Berlin, Germany, pp. 25-33, 2011.
- [5] van Pinxteren, Y., Geleijnse, G., Kamsteeg, P.. *Deriving a recipe similarity measure for recommending healthful meals*, Proceedings of the 16th international conference on Intelligent user interfaces. IUI '11, New York, NY, USA, ACM (2011) 105–114.
- [6] Robin Burke. *Hybrid Web Recommender Systems*, The Adaptive Web (Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl, eds.), LNCS 4321, Springer, 2007, pp. 377-408.
- [7] Ruth H. Keogh. *Allowing for never and episodic consumers when correcting for error in food record measurements of dietary intake*, Biostatistics (2011), 12, 4, pp. 624–636.