Les systèmes de coaching : recommandation alimentaire automatique pour un changement de comportement à long terme

Jules Vandeputte

▶ **To cite this version:**

Jules Vandeputte. Les systèmes de coaching : recommandation alimentaire automatique pour un changement de comportement à long terme. Intelligence artificielle [cs.AI]. Université Paris-Saclay, 2023. Français. NNT : 2023UPASB019 . tel-04109887

HAL Id: tel-04109887
https://pastel.hal.science/tel-04109887

Submitted on 30 May 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THESE DE DOCTORAT

NNT : 2023UPASB019

université PARIS-SACLAY

Les systèmes de coaching : recommandation alimentaire automatique pour un changement de comportement à long terme

Investigating food recommendation for long-term behaviour change: Introducing the coaching framework

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 581, Agriculture, alimentation, biologie, environnement et santé (ABIES)
Spécialité de doctorat : Informatique appliquée
Graduate School : Biosphera. Référent : AgroParisTech

Thèse préparée dans l'UMR **MIA Paris-Saclay** (Université Paris-Saclay, AgroParisTech, INRAE), sous la direction de **Antoine CORNUEJOLS**, Professeur, la co-direction de **Nicolas DARCEL**, Maître de Conférences (HDR)

Thèse soutenue à Paris-Saclay, le 3 avril 2023, par

Jules VANDEPUTTE

Composition du Jury

Membres du jury avec voix délibérative

Benoît GIRARD Directeur de Recherche, CNRS (Sorbonne Université)	Président
Armelle BRUN Professeure, Université de Lorraine	Rapporteur & Examinatrice
Chantal JULIA PU-PH, Université Sorbonne Paris-Nord	Rapporteur & Examinatrice
Jérémie MARY Chercheur, CRITEO	Examinateur
Sabrina TEYSSIER Chargée de Recherche, INRAE (Université Grenoble-Alpes)	Examinatrice

université PARIS-SACLAY

ÉCOLE DOCTORALE
Agriculture, alimentation, biologie, environnement, santé (ABIES)

Titre : Les systèmes de coaching: recommandation alimentaire automatique pour un changement de comportement à long terme.

Mots clés : Systèmes de recommandation, Prise de décision répétée, Habitudes alimentaires, Recommandation nutritionnelle

Résumé : De nos jours, la prise de décision se fait de de plus en plus en interaction avec une machine notamment via les algorithmes de recommandation. Ce travail de thèse vise à utiliser les outils développés dans le domaine des systèmes de recommandation afin d' accompagner un utilisateur dans un processus de modification de ses habitudes de consommation. Ainsi nous considérons le changement de comportement de l'utilisateur comme l'objectif de la recommandation, et appelons cette tâche de recommandation "coaching". L'objectif est d'explorer la manière de concevoir un tel système. Pour ce faire, nous proposons un modèle de l'interaction utilisateur-système, sous la forme d'un jeu itéré à deux joueurs. Nous explorons ensuite, via une étude formelle de ce modèle, les politiques de recommandation possibles, et leurs caractéristiques. Nous mettons en évidence l'importance de la personnalisation, et l'intérêt des stratégies non-myopes. Dans un second temps, nous étudions ce problème dans le contexte particulier de la recommandation alimentaire. En effet, les habitudes alimentaires jouent un rôle prépondérant sur la santé. Nous explorons donc l'applicabilité d'un tel système dans le monde réel et montrons l'importance pour l'acceptabilité des proposition du système, de l'implication de l'utilisateur dans l'élaboration des recommandations. Enfin, nous nous intéressons à l'introduction de données contextuelles dans l'évaluation du comportement utilisateur. Nous proposons une méthode originale basée sur la recommandation de cycles de consommations, pour contourner les limitations intrinsèques des utilisateurs.

Title : Investigating automated food recommendation for long-term behaviour change: Introducing the coaching framework.

Keywords : Recommender systems, Repeated decision-making, Eating habits, Food recommendation

Abstract : Nowadays, decision-making is increasingly computer-driven, because of recommendation algorithms. This thesis aims to use the tools developed in the field of recommender systems to accompany users in the change of their consumption habits. Thus we consider the change in user behaviour as the aim of recommendation and define this task as "coaching". The objective is to explore how to design such a recommendation system. To do so, we propose a model of the user-system interaction in the form of a two-player iterated game. Then we explore the possible recommendation policies and their characteristics through a formal study of this model. We outline the importance of personnalisation and the interest of non-myopic recommendation strategies. We study this problem in the particular context of food recommendations, as dietary habits play a major role in health, and healthier eating habits are key for public health policies. We, therefore, explore the applicability of such a system in the real world. In particular, we show the importance of user's involvement in recommendations' formulation on their acceptability. Finally, we focus on the introduction of contextual data in the evaluation of user behaviour. We propose an original method based on consumption cycles recommendation, to circumvent users limitations.

Résumé en Français

Contexte général

Au cours de l'histoire récente, l'émergence et le développement des technologies de l'information ont à plusieurs reprises bouleversé la façon dont les Hommes prennent des décisions. Notamment, il est devenu courant d'avoir à choisir, sur des plateformes en ligne notamment, parmi de très nombreuses propositions de biens ou de services. Pour divers items, tels que des films, des morceaux de musiques, des vêtements ou des contenus sur les réseaux sociaux, les avancées récentes dans la collecte et le stockage des données ont rendu possible, pour la plupart d'entre nous, l'exploration d'un immense champ des possibles. Il est même fréquent que l'examen de l'ensemble des items soit impossible. Par exemple, la plateforme de partage de vidéos en ligne *YouTube.com* comptabilise plus de 500 nouvelles heures de vidéos ajoutées chaque minute [1], ce qui revent à 80 années de contenu ajouté quotidiennement sur la plateforme. Le site de e-commerce *Amazon.com*, quant à lui, totalise plus de 350 millions de produits disponibles à la vente [2]. De fait, ces immenses quantités d'information ne peuvent être appréhendées par un être humain, et faire un choix parmi de si nombreuses possibilités s'avère souvent particulièrement ardu. Cette difficulté à choisir de manière éclairée parmi de nombreuses alternatives est le marqueur d'un phénomène bien connu : le problème de la *surcharge informationnelle*.

L'une des solutions ayant émergé pour faire face au problème de la *surcharge informationnelle* est le développement et l'utilisation d'outils connus sous le nom de "Systèmes de recommandation" (SR) [3]. Le but des SR est de filtrer l'ensemble du champ des possibles, afin d'en dégager une liste des items les plus pertinents. Ces derniers sont appelés les *recommandations* du système. L'idée est qu'en proposant à l'utilisateur du SR un nombre réduit d'items, leur comparaison est facilitée, et l'utilisateur devient capable de faire un choix éclairé, contournant donc le problème de la surcharge informationnelle. Un point essentiel pour l'efficacité de

telles méthodes est la notion même de *pertinence* des items. Idéalement, les recommandations proposées devraient être représentatives du champ des possibles, ainsi qu'adaptées pour chaque utilisateur du SR. Depuis maintenant plusieurs années, les systèmes de recommandations se sont imposés comme un champ de recherche à part entière, et les travaux de recherche se sont en partie concentrés sur le fait de trouver les meilleures recommandations possibles, soit la meilleure sélection d'items pour un utilisateur donné à un instant donné. De nombreuses mesures de l'intérêt de l'utilisateur pour un item existent, qui dépendent du type de données considérées (notations explicites, temps d'écoute ou de visionnage ou encore taux de clics). En découle une grande variété d'approches possibles du problème de la recommandation. La mise au point d'algorithmes de recommandation efficaces met ainsi à profit de nombreuses méthodes issues de disciplines connexes, telles que l'intelligence artificielle, l'apprentissage automatique ou la fouille de données. La combinaison du potentiel commercial des SR et de l'intérêt de la recherche académique pour le sujet a mené à des avancées rapides en terme de qualité des recommandations. De plus, étant donné les effets positifs observés des SR, sur les activités commerciales notamment, ces derniers se sont rapidement diffusés sur de nombreuses plateformes en ligne, et sont devenus des outils avec lesquels il est commun d'interagir.

Impact des recommandations sur le comportement

La large diffusion des SR et leur omniprésence en ligne, visant à faciliter les choix des utilisateurs de nombreuses plateformes, pose la question de l'impact des SR sur le comportement de ces derniers. En effet, en tant que dispositif facilitant la prise de décision, les SR ont de fait un impact sur les choix de leurs utilisateurs, et donc plus largement sur leur comportement. Par ailleurs, ils leur permettent de découvrir et d'expérimenter de nouveaux items, ce qui peut éventuellement impacter leurs goûts et leurs préférences. De cette observation découlent deux questions:

Premièrement sur la capacité des algorithmes de recommandation à être pertinents dans un environnement évolutif. En effet, une recommandation qui qui aurait été intéressante pour un utilisateur donné à un instant donné peut devenir complètement inappropriée, pour peu, par exemple, que les intérêts de l'utilisateur aient changé.

Dans un second temps, sur l'importance de l'effet des recommandations sur l'évolution du comportement des utilisateurs. En effet, si de l'interaction avec différents items

découlent des changements dans les intérêts de l'utilisateur, et que le rôle d'un SR est de favoriser l'interaction de l'utilisateur avec certains items, la question de l'impact des SR sur les intérêts de l'utilisateur mérite d'être posée.

De fait, la question de l'impact des SR sur les habitudes de consommation ou la diversité de l'information est une préoccupation majeure de la littérature. Il existe deux canaux par lesquels les SR peuvent impacter le comportement utilisateur. Tout d'abord, en recommandant un item qui finit par être consommé par l'utilisateur, le SR impacte sur le court terme le comportement de celui-ci. Mais les impacts peuvent aussi être considérés à plus long terme. Prenons l'exemple d'un utilisateur qui écouterait de la musique sur une plateforme de streaming audio. Un morceau lui est alors recommandé, qu'il ne connaissait pas, et qui lui plaît particulièrement. Il est alors possible et même probable qu'il cherche à réécouter ce morceau par la suite. Il pourrait même par ce biais découvrir un nouvel artiste ou un nouveau genre qu'il apprécie. Cet exemple simple illustre comment un SR peut impacter le comportement d'un indvidu sur le long terme. Ou pas, l'utilisateur pouvant tout aussi bien juste passer le morceau en question. Les impacts des SR sur le comportement utilisateur sont de plus en plus étudiés, en particulier dans la recherche académique. Cependant la majorité des travaux sont plutôt focalisés sur les impacts négatifs de ces systèmes. En effet cette problématique a notamment émergé suite à la popularisation du concept de bulles de filtres, désignant des cas ou les utilisateurs font face, potentiellement à cause de recommandations, à des contenus très similaires entre eux, ne représentant qu'une part très limité des informations ou items existants. Le manque de diversité dans les recommandations, découlant de ce phénomène, a été largement étudié [4]. Notamment, le sous-domaine de la recommandation d'actualités et les enjeux autours des impacts sur la démocratie ont été particulièrement explorés [5]. Pour les mêmes raisons les impacts des SR ont majoritairement été considérés à un niveau global, de nombreux travaux tentant de comprendre comment différents algorithmes induisent différent changements de comportement dans de grands groupes d'utilisateurs. Une question largement étudiée est l'impact des SR sur la diversité des items effectivement consommés par les utilisateurs. Il a notamment été montré que les algorithmes étaient souvent biaisées, et qu'en résultaient des recommandations favorisant les items les plus populaires, et ce pour l'ensemble des utilisateurs [6].

Mais la question de l'impact des recommandations à un niveau plus particulier, et

notamment de comment un SR interagit avec les habitudes de consommation et les préférences de chaque utilisateur mérite également d'être posée. Compte tenu des méthodes et algorithmes de recommandation existants, nombre de recommandations sont, au moins partiellement, guidées par les préférences des utilisateurs. Or, comme nous l'avons vu plus tôt avec l'exemple de la recommandation de morceaux de musique, les recommandations peuvent, en retour, impacter les habitudes des utilisateurs. Il notamment été montré que les SR peuvent impacter le comportement individuel des utilisateurs, par exemple en réduisant la diversité des contenus consommés par un utilisateur particulier [4]. D'autre part, de récents travaux dans le domaine de la recommandation musicale ont montré que les SR pouvaient avoir une influence sur les préférences des utilisateurs [7]. Dans cette thèse, nous nous penchons particulièrement sur ce type d'impact, au niveau individuel. Plus particulièrement, nous nous intéressons à l'impact des SR sur les habitudes sur le long terme et à la conception d'algorithmes de recommandation ayant pour objectif d'accompagner un utilisateur donné dans un processus de modification de ses habitudes de consommations.

Motivations

L'objectif de ce travail de thèse est de tirer profit de l'impact des systèmes de recommandation sur les comportements de leurs utilisateurs, afin de s'en servir comme d'un outil pour faciliter le changement d'habitudes de consommation. En effet, dans de nombreux domaines, notamment celui de la santé, nous pouvons aspirer à changer nos habitudes afin de nous rapprocher d'un objectif, comme celui de manger plus sainement, de faire plus d'activité physique, ou de réduire notre empreinte environnementale. Néanmoins, il peut être difficile de mettre effectivement en place, dans notre vie quotidienne, des changements en accord avec ces aspirations, notamment car notre motivation est souvent fluctuante. Notre postulat, dans cette thèse, est de mettre à profit les systèmes de recommandation, et leurs interactions avec les choix et habitudes de leurs utilisateurs, afin d'aider ces derniers à faire des choix éclairés et à finalement atteindre leurs objectifs.

Bien que l'approche développée dans le présent manuscrit soit générale et applicable à de nombreux domaines, nous nous focaliseront ici sur le problème du choix alimentaire. L'influence de l'alimentation et de la nutrition sur le bien-être et la

santé est en effet connue depuis très longtemps, et de fait aujourd'hui, de nombreuses politiques de santé publique se focalisent sur l'alimentation. Notamment, de nombreuses administrations publiques ont produit, dans divers pays du monde, des recommandations nutritionnelles ainsi que des programmes visant à promouvoir une alimentation saine. Les exemples de telles politiques sont nombreux [8, 9]. Bien que l'efficacité de ce genre de programme ait été démontrée, des efforts sont encore à faire pour arriver à des systèmes d'alimentation sains et durables. De plus, au niveau individuel, la volonté d'un consommateur à manger de façon plus saine peut être entravée, d'une part par ses connaissances limitées en nutrition et, d'autre part, par des habitudes de consommation fortement ancrées. Comte tenu de ces enjeux, le domaine de la nutrition semble être particulièrement adapté au développement d'un SR visant à impacter les habitudes de consommation. En outre, la nature répétée ainsi que la fréquence du choix alimentaire ouvre potentiellement la porte à des impacts importants sur le comportement utilisateur.

Définition du problème

Dans ce travail de thèse, nous nous intéressons au problème de l'élaboration de systèmes de recommandation encourageant efficacement leurs utilisateurs à modifier leur comportement, et plus particulièrement leur comportement alimentaire afin de promouvoir une alimentation saine. En effet, les systèmes de recommandation étant de plus en plus omniprésents dans notre vie quotidienne, notre hypothèse est qu'ils pourraient être utilisés comme levier pour accompagner des consommateurs dans une dynamique de modification de leurs habitudes. Les systèmes de recommandation peuvent en effet affecter le comportement de deux façon. Tout d'abord ponctuellement, lorsqu'un utilisateur accepte une recommandation donnée. Mais également à plus long terme, puisque nous l'avons vu, la recommandation peut potentiellement affecter les habitudes d'un utilisateur.

L'objectif du présent travail est de tirer parti de ce second levier pour induire chez un utilisateur des changements de comportement, notamment alimentaire, sur le long terme, afin de le guider vers des habitudes de consommation plus saines. Aussi contrairement à la majorité des solutions de l'état de l'art dans le domaine des systèmes de recommandation, nous nous penchons sur les effets à long terme des recommandations, plutôt que sur leur acceptation immédiate. Ce prisme soulève

différentes questions, tant sur la production des recommandations que sur leur communication à l'utilisateur. Tout d'abord, cela nécessite d'être capable de capturer les effets des recommandations sur le long terme, et donc d'interagir de façon répétée avec les utilisateurs. Ensuite, amener un réel changement dans le comportement de l'utilisateur nécessite que ce dernier soit volontaire pour ce changement, et impliqué dans le processus. Compte tenu de ces spécificités, nous définissons ici le problème de recommandation correspondant comme un problème de *coaching*.

Définition du coaching

Une interaction de coaching, d'après [10], nécessite au moins deux participants : d'un côté le coach, et de l'autre son élève, considéré comme mature, motivé, volontaire, et engagé dans une relation d'apprentissage avec un "facilitateur" (le coach) dont le rôle est d'aider l'élève à atteindre ses objectifs. Cette définition, bien que générale, décrit bien l'objectif de la présente thèse, avec dans le rôle du "facilitateur", le système de recommandation. Notre objectif est en effet de tirer parti des systèmes de recommandation pour accompagner un utilisateur dans un dynamique de modification de son comportement. Ainsi, nous explorons dans cette thèse la question de la conception de *systèmes de coaching*, c'est à dire de systèmes automatisés fournissant à leurs utilisateurs des recommandations promouvant un changement de comportement. De plus compte tenu de cette définition, nous considérons les utilisateurs comme conscients de leur objectif final, mais pas nécessairement capables d'évaluer correctement chacun de leur choix au regard de cet objectif, d'où l'intérêt du coach.

Questions traitées dans cette thèse

Étant donné la définition ci-dessus, et le problème général de la recommandation pour la modification des habitudes, nous pouvons formuler plusieurs questions de recherche, auxquelles la présente thèse a pour objectif de répondre. La première de ces questions est la suivante:

- **Q1:** *Comment concevoir un cadre de recommandation qui promeuve le changement de comportement de ses utilisateurs sur le long terme?*

Comme nous l'avons déjà mentionné, bien que traitant un problème de recomman-

dation général, la présente thèse est motivée par le problème de la recommandation nutritionnelle, et les enjeux de santé publique associés. Dans ce contexte, il apparaît essentiel de considérer l'applicabilité et la faisabilité des solutions proposées dans le domaine alimentaire. D'où la question suivante:

- **Q2:** *Comment devrait être pensé un système de coaching automatique pour faire des recommandations acceptables dans le domaine de l'alimentation?*

À partir du problème soulevé par la mesure de la qualité nutritionnelle d'un comportement, on peut entrevoir la difficulté de trouver des métriques à la fois simples et pertinentes pour évaluer l'intérêt d'un choix donné au regard d'un objectif spécifique. En particulier, les notions de contexte et d'historique de consommation jouent souvent un rôle essentiel. Dans ce cadre, il apparaît capital pour un système de recommandation visant une modification du comportement de prendre en compte ces notions dans l'évaluation du comportement. D'où la question:

- **Q3:** *Comment un système de coaching automatique peut inclure le contexte et les dynamiques temporelles dans l'évaluation de ses recommandations?*

Contributions

Ce travail de thèse à débouché sur trois principales contributions scientifiques :

- **Première contribution: Création d'un cadre conceptuel de recommandation pour la modification des habitudes, nommé *coaching*.** Nous nous sommes intéressés au problème de la recommandation répétée avec un objectif de modification des habitudes de consommation. Nous avons proposé un modèle d'interaction entre le système de recommandation et son utilisateur, sous la forme d'un jeu itéré à deux joueurs. Nous avons exploré la question de la production de recommandations visant le changement d'habitude, et formalisé le problème comme un problème d'apprentissage par renforcement. De cette étude formelle du problème, nous avons dérivé diverses stratégies de recommandation pour les systèmes de coaching, et les avons testées dans le cadre de simulations basées sur des données réelles de consommation.

- **Deuxième contribution: Étude de l'applicabilité du cadre du coaching au problème de la recommandation alimentaire.** Avec pour objectif

d'appliquer le formalisme des systèmes de coaching à la recommandation nutritionnelle, nous avons mené deux expérimentations. La première visait à tester une solution possible au problème dit du *cold-start* pour les systèmes de coaching, basée sur l'extraction de valeurs de substituabilité entre aliments à partir de données de consommation. La seconde visait à tester différentes modalités d'interaction homme-machine et leur pertinence dans le contexte des systèmes de coaching.

- **Troisième contribution: Conception d'une méthode permettant de produire des recommandations pertinentes dans le cas d'une évaluation contextuelle du comportement utilisateur.** Nous avons étudié la question du contexte dans le cadre du coaching, et en particulier dans l'évaluation du comportement utilisateur. Nous avons mis en évidence le problème de la représentabilité pour l'utilisateur du comportement cible visé par le système de coaching. Pour répondre à ce problème, nous avons proposé une méthode basée sur des cycles de consommation pour contourner les limitations intrinsèques des utilisateurs. Compte tenu de la complexité de calcul associée à la découverte de tels cycles de consommation, nous avons proposé une heuristique, permettant d'approcher la solution optimale. Nous avons également proposé une heuristique pour la production de recommandations visant à amener un utilisateur vers un comportement donné.

Remerciements

Je tiens tout d'abord à remercier mes encadrants, Antoine Cornuéjols, Nicolas Darcel et Christine Martin, qui ont su m'accompagner et me guider tout au long de mon travail de recherche. Merci d'avoir imaginé ce projet de thèse, et de m'avoir fait confiance pour le mener à bien. Merci également pour nos échanges et discussions, qui m'ont beaucoup appris et permis de découvrir de nouveaux horizons scientifiques. Merci Antoine pour ton investissement et tes conseils au cours de la rédaction du manuscrit. Merci Nicolas pour ta bonne humeur et tes remarques, toujours constructives. Merci Christine pour tes conseils qui m'ont permis de prendre un recul bienvenu sur mon travail. Merci également à Faboien Delaere pour son investissement dans cette thèse et ses remarques et points de vue, toujours très pertinents.

Merci également à Armelle Brun et Chantal Julia d'avoir accepté d'être rapporteurs de ma thèse, et d'avoir pris le temps de lire dans le détail le présent manuscrit. Un grand merci pour leur remarques particulièrement pertinentes, notamment le jour de la soutenance. Merci aussi aux autres membres du jury, Sabrina Teyssier, Jérémie Mary et Benoît Girard, pour leurs questions et observations.

Je tiens aussi à remercier l'ensemble des membres du laboratoire MIA-Paris pour leur acceuil chaleureux. Le début de cette thèse a été un peu troublé par les périodes de confinement successifs, mais j'ai trouvé suite à ça une vraie vie de labo, que j'ai apprécié partagé avec vous tous. Un merci particulier aux doctorants du labo pour leur bonne humeur et pour les soirées que nous avons passé ensemble. Vivement le prochain karaoké.

Je souhaite également remercier ici mes colocataires : Michel, Simon, Eliott, Charles, Bertrand et Matthéo. Vous m'avez supporté pendant ces trois années (à tour de rôle), et ce malgré des périodes difficiles, et c'est aussi grâce à cette ambiance familiale et bon enfant que je retrouvais en rentrant à la maison tous les soirs que j'ai apprécié ces trois ans. Vous avez tous contribué à votre manière à l'achèvement

de ce travail, et même si je vous ai épargné la relecture du manuscrit, il vous doit beaucoup. J'ai trouvé avec vous à la fois un super cadre de vie, des amis pour les moments de décompression, et un soutien dans les périodes les plus tendues. Un immense merci pour ça ! Michel le prochain c'est toi !

Merci à mes copains qui ont fait le déplacement pour la soutenance et le pot, je suis heureux d'avoir pu partager l'aboutissement de ces trois années avec vous.

Merci à mes parents, qui sont toujours de bon conseil, et qui ont su m'épauler au moments où il le fallait. Merci de leur soutien sans faille. Merci papa de m'avoir donné envie de faire une thèse et de m'avoir conseillé dans mes moments de doute, et merci maman de m'avoir écouté et rassuré lors de mes moments de stress. Je sais que tu sais ce que c'est. Merci à mon frère Emile, tu es sans doute celui qui partage l'expérience la plus proche de la mienne, et pouvoir en parler avec toi a été vraiment appréciable. J'espère que c'est aussi le cas pour toi et que j'aurai droit à un petit mot dans ta thèse. Merci à ma sœur Lucie, pour tous les moments de rigolade partagés ensemble, je suis super content que tu aies trouvé ce qui te plaît et que tu t'y investisses comme tu le fais. Merci Nicolas et Sylvie pour votre présence et vos encouragements à chaque fois que je venais vous voir. Et plus généralement merci à toute ma famille pour son soutien.

Enfin merci Yoluène de m'avoir accompagné, soutenu et supporté pendant ces trois ans. Je sais que ça n'a pas du être facile tous les jours, mais je suis vraiment heureux de t'avoir à mes côtés, tu m'apportes tellement de joie, merci. Et je te souhaite de tout mon cœur d'avoir un jour l'occasion de mener à bien tes propres projets de recherche.

Cette thèse a été financée dans le cadre du projet ANR SHIFT.
(ANR-18-CE21-0008)

Contents

1. Introduction

1.1. General Context

In recent history, the emergence and development of information technologies led to numerous disruptions in how humans make their decisions. In particular, it became common to have to make choices among vast collections of items or services on online platforms. For many items as diverse as movies, music tracks, clothes or relatives on social media, recent advances in data collection and accessibility made it possible for most of us to explore a tremendous number of possibilities. Frequently it is not even possible to examine the whole item space. For example, consider that 500 hours of video are uploaded every minute on *YouTube.com* [1], i.e. more than 80 years of content are made available daily on the platform. On the *Amazon.com* market place more than 350 million products are available [2]. It appears that this immense quantity of information cannot be handled by a human being and that making a choice among such a massive variety is particularly arduous. This difficulty to choose wisely among many possibilities is the indicator of a well-known problem in human decision-making, referred to as the *information overload* problem.

A solution that emerged to overcome the *information overload* problem is the use of so-called *recommender systems* (RS) [3]. The task of RS is to filter the entire item space to provide their users with the most interesting items. These are known as the recommendation(s) of the system. The idea is that, by proposing a reduced number of items to the user, their comparison is facilitated. In other words, the *information overload* problem is reduced, and the user should be able to make an informed choice. The key point for such systems to be efficient is the very notion of "interesting items". Ideally, the proposed recommendation(s) should be representative of the item space and adapted to each particular user. With the emergence of RS as a discipline on its own, researchers focused on the problem of finding the best recommendations, that is, the best selection of items for a given user at a given time. A wide variety of measures exist for the user's interest in an item depending on the type of data used (ratings, click rate,

watch time, etc.) and, consequently, a wide variety of approaches to the recommendation problem. Designing efficient recommendation algorithms involves many techniques from related fields such as artificial intelligence, machine learning or data mining. Both the commercial potential and the interest of researchers in the field led to a wide development of RS and a rapidly increasing performance of algorithms in terms of recommendation quality. As they showed to have a substantial impact on online business activities, RS spread over the web and became very common for internet users to interact with.

1.1.1. Impact of recommendation on behaviour

The widespread use of RS and their pervasiveness on online platforms to facilitate users' choices raise questions about the corresponding impacts on the latter's behaviour. Indeed, if RS aim to facilitate decision-making, they inherently impact their users' choices and, more generally, their behaviour. Moreover, by discovering and experiencing new items, users' beliefs and interests may change, which poses two major questions.

First, it interrogates the capacity of the recommendation algorithms to learn in an evolving environment. Indeed, a recommendation that would have been interesting for a given user at a given time may become utterly irrelevant if the latter's interests have changed.

Second, it highlights the importance of the recommended items in the evolution of user behaviour. Indeed, if users' beliefs change as they experience various items, the role of the RS in such a change merits being questioned. For example, the impact of RS on consumption habits or information diversity is a major concern in the literature. In fact, a RS can directly impact users' behaviour by recommending an item that the user eventually purchases, for example. Nevertheless, it can also have longer-term impacts. For example, consider a user listening to music on a streaming platform, and a song is recommended to him or her that he/she did not know and that he/she particularly appreciates. Then he/she may listen to the song again in the future or even discover throughout this song a new artist that he/she likes. This simple example illustrates how a RS may impact a user's behaviour in the long term. Or may not, as the user could also just skip the song.

The impact of RS on user behaviour is increasingly studied and investigated, particularly in academic research. However, the majority of the works are focused on negative impacts. Indeed, the first concern about the impact of RS on user behaviour arose from the concept of "filter bubbles", which is the fact that a user can be trapped, possibly due to recommendation, in a subspace of the item space where all the information he

or she gathers is similar. Moreover, the observed lack of diversity in recommendations ensuing from it is questioned in the literature [4]. In particular, the sub-field of news recommendation, and the corresponding stakes about impacts on democracy, were specially studied [5]. The same reasons led to considering the impact at a global level, with works trying to understand how RS design induces changes in large groups of users. A widely studied point is the impact of RS on the diversity of the items consumed by the users. In particular, it has been shown that RS are biased towards the most popular items and tend to recommend them to all the users [6].

But one can also consider the impact of the recommendation on a single user and how a RS interacts with his consumption habits and preferences. Given the current design of the majority of RS algorithms, the recommendation is at least partly driven by the preferences of the user. However, as illustrated with the soundtrack example, the recommendation may also, in return, impact the habits and preferences of the user. It has been shown that RS can impact the individual behaviour of users, for example by narrowing the diversity of the consumed content [4]. Moreover, recent work in the music recommendation domain have shown that RS can impact users' preferences [7]. In this thesis work, the focus is made on this type of impact at the user level. We are particularly interested in the impact RS can have on habits in the long term. More precisely, the following work investigates RS design with the objective of accompanying a given user through a behaviour change process.

1.2. Motivation

This work aims to take advantage of the impacts of RS on users' behaviour and to use them as a tool to facilitate the behaviour change of a user. Indeed, in many domains, including health and self-care, people may aspire to initiate a change in their habits and reach an objective such as eating healthier, being more active, or reducing their environmental impact. However, these aspirations may be difficult to realize, as one's motivation can fluctuate over time. Our postulate is to use RS and their interactions with users' choices and habits to help users in making informed choices and eventually reach their objectives. Although a general approach that we strongly believe to be relevant in various application cases, we mainly focus in this work on the problem of promoting healthy eating. Humans have known from long ago the role of nutrition and food in welfare and healthiness. As such, many questions about public health involve questions about nutrition. One marker

of this is the interest of public administrations in elaborating guidelines and programs designed to impact populations eating behaviours and eating habits. Examples of so-called public nutritional guidelines are numerous and can be found worldwide [8, 9]. Even though they have proven to have a positive effect on public health, improvements are still to be made to ensure healthy and sustainable food systems. Moreover, at the individual level, a consumer's will to eat healthier may collide with his/her limited knowledge of food items' healthiness or firmly rooted consumption habits. Due to these stakes and their capital importance, the task of healthy eating appears to be well adapted to develop RS to impact habits. In addition, the repeated nature of food choice and its frequency opens the room for a significant impact on users' behaviour.

1.3. Problem definition

In this thesis, we are mainly interested in the problem of building a recommendation system encouraging efficiently a user to modify his or her behaviour, and more particularly his or her eating behaviour, in order to promote healthy eating. Indeed, recommendation systems are increasingly present in our everyday life, and we assume that they could be used as a lever to accompany their users in a dynamic of behaviour change. As presented in 1.1, recommendation systems can affect behaviour in two different ways: the first is ad hoc. It relies on the immediate change induced by the acceptance of a recommended item. The second is lasting and is due to the change in user habits that the recommendations may induce.

The aim of this work is to use this second lever to induce persistent changes in the user's behaviour, particularly eating behaviour, guiding him or her towards healthy eating habits. Thus, conversely to the majority of state-of-the-art solutions in recommendation systems, we focus more on the long-term effects of the recommendations rather than on their immediate acceptance.

This focus raises different questions, both on the production and the distribution of the recommendations. Firstly, it requires being able to capture the long-term effects produced by the recommendations and so to interact repeatedly with the same user. Secondly, to bring about a real behaviour change, the user must be willing to change and must be involved in the recommendation process. Regarding these particularities, we define this recommendation problem as the *coaching* problem.

1.3.1. Definition of coaching

A coaching interaction involves at least two participants. First, the coach, and second the learner, which is "perceived to be a mature, motivated, voluntary, and equal participant in a learning relationship with a facilitator whose role is to aid the learner in the achievement of his or her primarily self-determined learning objectives" according to [10]. This definition, although general, describes well the objective of this thesis work, with the recommendation system in the role of the so-called "facilitator". Indeed, far from manipulating the user, our objective is to take advantage of recommendation systems and their effects on consumption habits to accompany a user in a behaviour change process. Therefore, we explore the design of *coaching systems*, which are automated systems providing a user with recommendations promoting behaviour change. Moreover, regarding this definition, we consider the user as aware of his/her final objective, but not necessarily able to judge wisely the choices that lead towards this objective. Hence the interest of the coach.

1.3.2. Questions addressed in this thesis

Thus, regarding this definition, we formulate different questions that emerge from our main problem and that are addressed in this thesis. Regarding the specificity of the considered recommendation problem, the first question we address is:

- **RQ 1:** *How to design a recommendation framework that promotes long-term behaviour change of its users?*

As previously mentioned, although investigating a general problem of recommendation, this thesis is motivated by the issue of healthy eating promotion and the corresponding stakes of public health. In this context, it appears essential to consider our proposed solution's feasibility and applicability to the food recommendation domain. Thus, the second question addressed in this thesis is:

- **RQ 2:** *How should be designed an automated coaching system to make acceptable recommendations in the food domain?*

Starting from the problem of healthiness measure in food consumption, we can assess the difficulty of finding simple yet representative metrics evaluating one's behaviour regarding a specific objective. In particular, considering notions of history and context is

often critical, although leading to greater complexity. In this context, it appears crucial for a recommendation system targeting behaviour change to support various elements, including temporal dynamics, when evaluating the possible recommendations. So the last question tackled in this work is :

- **RQ 3:** *How could an automated coaching system incorporate context and temporal dynamics in the evaluation of its recommendations?*

1.4. Contributions

This work has led to three main contributions, which are the following:

- **First contribution: Creation of a framework to address recommendations for behaviour change, known as the coaching framework.** We considered the problem of repeated recommendations with the objective of behaviour change. We proposed a model of the interaction between the RS and the user as an iterated two-player game. We investigated the task of making recommendations for user behaviour change and modelled it as a reinforcement learning problem. Moreover, we derived from the formal study of the problem several possible recommendation strategies for coaching systems and tested them on a simulated recommendation task based on real consumption data.

- **Second contribution: Investigation of the applicability of the coaching recommendation framework for healthy food recommendation.** With the objective of using coaching for healthy food recommendations, we conducted two real-world experiments. The first investigated a possible solution to the *cold-start* problem in coaching by testing against real users the likelihood of possible food substitutions mined from consumption data. The second tested different modalities of human-computer interaction in the context of automated coaching.

- **Third contribution: Conception of a method to discover pertinent coaching recommendations in the case of a contextual evaluation of user behaviour.** We proposed to study the question of context in the coaching framework and, in particular, the contextual evaluation of user behaviour. We highlighted the problem of the representability for the user of the behaviour targeted by the

coaching system. We proposed a method based on cyclic behaviour recommendations to circumvent the limitations of the user. Given the complexity of mining for such cyclic behaviours, we proposed a heuristic algorithm to approach the solution. We also proposed a heuristic algorithm to produce recommendations leading a user towards a target behaviour.

1.5. Thesis structure

The remainder of this thesis is organised as follows :

- Chapter 2 presents a review of the literature on related problems. Moreover, we place the concept of *coaching systems* within the landscape of the existing literature. We show that the coaching problem, as defined, encompasses questions from several domains, thus we propose a synthesis of diverse approaches from different research communities that are all capturing a part of the problem.

- Chapter 3 presents our first contribution, which is the framework we developed for the recommendation for behaviour change, namely the *coaching framework*. We formalize the problem of automated coaching and model it as a two-player iterated game. We propose a general theoretical solution to the recommendation problem and derive from it approximate solutions. We test these solutions on a simulated recommendation task and deduce from the results some key characteristics of efficient recommendation strategies for coaching.

- Chapter 4 presents our second contribution. We first discuss the approaches of the literature on healthy food recommendation and the applicability of the coaching framework in this context. We then present the methodology and results of the two real-world experiments we conducted. Finally, we discuss the evaluation of healthiness in food recommender systems and consider two possible approaches.

- Chapter 5 presents our third contribution. We investigate the introduction of contextual data in the coaching framework and in the evaluation of user behaviour. We formalize the problem of finding the best behaviour for a user in this setting and propose a general solution. We also propose a heuristic to approach the solution in an application case. We then discuss the question of making recommendations towards a given target behaviour and propose a heuristic algorithm.

- Chapter 6 concludes and discusses perspectives and future research directions for this work.

1.6. Scientific publications

This work led to three publications.
Peer-reviewed French national conferences:

- **Jules Vandeputte**, Antoine Cornuéjols, Nicolas Darcel, Fabien Delaere, Christine Martin. (2021). *Le coaching: un nouveau cadre pour la recommandation automatique en vue de modifications durables du comportement.* In CNIA 2021: Conférence Nationale en Intelligence Artificielle (pp. 44-51).

Peer-reviewed international conferences with proceedings:

- **Jules Vandeputte**, Antoine Cornuéjols, Nicolas Darcel, Fabien Delaere, Christine Martin. (2022, May). *Coaching Agent: Making Recommendations for Behavior Change. A Case Study on Improving Eating Habits.* In Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems (pp. 1292-1300).

International peer-reviewed journal (*accepted for publication*):

- **Jules Vandeputte**, Pierrick Herold, Mykyt Kuslii, Paolo Viappiani, Laurent Muller, Christine Martin, Olga Davidenko, Fabien Delaere, Cristina Manfredotti, Antoine Cornuéjols, Nicolas Darcel. *Principles and validations of an Artificial Intelligence-based recommender system suggesting acceptable food changes.* The Journal of nutrition

2. State of the art and related literature

As introduced in Chapter 1, the problem faced in this thesis is to find a way of making efficient recommendations for lasting behaviour change. Given that this question is both challenging from the theoretical side and interesting in diverse application domains, many distinct scientific communities from different disciplines worked on related problems. This leads to extensive but fragmented literature with many possible approaches. By presenting a review of the literature on the more meaningful related problems, the following chapter aims to give an overview of existing research, its links, and differences with the studied question. Moreover, the objective is to place the concept of automated coaching in the existing body of literature and to highlight the links that it draws between diverse concepts from machine learning to human decision theory.

As such, we are interested in the recommender systems literature that focuses on the problem of how to make recommendations from data. In fact, recommender systems (RS) appear to potentially have impacts on the habits of their users [11], and this could be used to promote useful behaviour [12]. These systems were mostly studied from an industry point of view, willing to maximize the consumption of the user. But we are also interested in the literature about RS with goals other than pure acceptance maximization. The fact that coaching is based on an interaction between a user and a coach or facilitator that gives advice leads to being interested in learning frameworks with two distinct agents. In particular, the inter-agent transfer learning problem seems to be of interest when considering a seasoned agent (i.e. the coach) interacting with a naive one (i.e. the user). Thus, we review the literature on inter-agent transfer learning, focusing on the particular teacher-student framework.

The scope of this thesis work is to study how recommendations can be made to have a lasting impact on user habits. That is how an automated system can lead to long-term user behaviour change. Given this, we explore the literature on two fields: persuasive technologies and intelligent tutoring systems, grouped under the name of technologies for behaviour change.

This chapter is organized as follows: In Section 2.1, we examine the classical recommender systems approach and the associated algorithms. In Section 2.2, we present the inter-agent transfer learning problem, the state-of-the-art solutions, and the literature on the so-called teacher-student framework. Section 2.3 presents an overview of technologies for behaviour change. Finally, Section 2.4 concludes on the presented literature and replaces the coaching problem in the outlined landscape.

2.1. Recommender systems

2.1.1. Recommender systems overview

Recommender system (RS) research emerged in the 1990's [13, 14] due to the rapid increase in data collections sizes. The objective was to help users to select the most meaningful data or items for them from these collections. Thus RS can be considered as filters that help a user to select the most relevant items. This is why we are interested here in this approach. Indeed, a coaching system's aim is to accompany a user towards a new behaviour. To do so, we can consider that the system has to filter all the possible actions for the user to recommend the ones that are the more relevant given the pursued objective. In this sense, the concept of coaching is closely related to RS. Indeed, the objective is to make *meaningful* and *personalized* recommendations to help a user in the task of changing his behaviour, which according to [14], are essential features of a RS.

2.1.1.1 Formal definition and framework

RS involve two agents that are interacting. First of all, the *recommender*, which role is to select an interesting item or short list of items and to recommend it to the second agent known as the *user*. Then the user may follow the recommendation and choose a recommended item. This phase is known as the *interaction* phase. The recommender faces three different data sources that can be used to generate a personalized recommendation. These data sources are the following :

- **Items.** Items are the purpose of RS, as they are the recommended objects. Items can be as diverse as movies [15], music tracks [16], lifestyles [17] or touristic activities [18]. In addition to the name of items, RS can consider meta-data associated with items, such as the genre of a movie or the price of a trip. These could be determinants

for the intrinsic utility of a given item for a user. It is important to notice that every item is associated with a cost, representing the effort made by the user to choose this item [14]. If a recommended item is accepted, this means that the benefit dominates the cost for the user. In other words, some recommendations are more acceptable for the user than others. The space of all possible items is denoted I, and every single item is denoted i, such as $i \in I$.

- **Users.** As stated, the users $u \in \mathcal{U}$ are the second type of agents that interact with the recommender and receive the recommendation, with $\mathcal{U}$ the set of all users. Given that personalization is a core feature of RS, gathering user data to make recommendations appears essential. The data gathered on the users form the user model [14], which is exploited to generate recommendations and is, as such, a key point of the RS. The data that feed this model can be acquired from the interaction with the user, such as behavioural data, or from preliminary investigations, such as user's attributes (e.g., age, gender, etc.)

- **Transactions or Interactions.** The objective of RS is to formulate recommendations and to propose them to the user. The user may then choose to follow the recommendation or not. Thus, in this phase, the RS and the user interact with each other. By doing so, the user generates feedback about his/her behaviour when facing a recommendation that will feed the user model of the RS. This feedback can be explicit, like, for example, the rating of a movie by a user, or implicit, that is to say, the RS collects it without the user's intervention. It can include, for example, the time spent on a web page or the items the user clicked on.

Therefore, a RS computes data from these three sources to make relevant recommendations. There are various algorithms to do so that rely on different approaches. We will develop these in the following section.

2.1.1.2 Classical recommendation approaches

As we have seen, there are three data sources for a recommender system: the users, the items, and the interactions. But, according to [3], we can classify these into two separated data types: the attributes information about users or items and the collected feedback on interactions. A recommendation approach is then defined by the data type it focuses on. Approaches that focus on the feedback are known as *collaborative filtering* approaches,

while approaches that focus on the attributes are known as *content-based*. Approaches where the system makes recommendations by leveraging task-specific knowledge are referred to as *knowledge-based*. Finally, all or some of these three can be combined to make more efficient recommendations in so-called hybrid recommendation systems. In the following, we will present these different approaches and their characteristics.

- **Content-based** methods focus their recommendations on attributes, generally of the items. In this approach, the system recommends items that are relevant to a user by considering his/her preferences in terms of content, that is, items that are similar regarding a set of attributes to the ones preferred by the user (See Figure 2.1). This can encompass both explicitly expressed preferences and historical data (i.e. the recorded feedback). The nature of these attributes and how they are obtained by the recommender system can be multiple. For example, the attributes can range from the genre of a movie [14] to the mood of a music track [19]. They can be generated either manually (e.g. content provider labelling of items) or automatically (e.g. extraction via natural language processing of the item description). In this approach, the RS builds a user-specific model that may be able to predict if a target item is likely to be appreciated by this particular user, given his/her registered preferences. This type of method is very effective when recommending new items for which no feedback data is available. Indeed, the only needed information to evaluate an item is extracted from the item itself. However, content-based methods are much less interesting when making recommendations for new users, as it is generally necessary to have an extensive user history to make accurate recommendations. Another drawback of content-based methods is the fact that the recommendations may seem *obvious* to the user, as they are made by considering only similar items to the ones already consumed by the user. Thus in this setting, the RS does not profit from the collective data generated by the community of all users.

- **Collaborative filtering** approach, contrastingly, exploits the community's power to recommend items. Indeed, this approach also uses the notion of similarity, but the similarity is computed based on all the feedback observed by the system. In the first implementation of such systems, each user was able to filter e-mails based on their interest to a selected group of other users [20], considered relevant by him/her. This is the principle of making recommendations based on the feedback of users with similar tastes, which is used and generalized in classic collaborative filtering

systems (See Figure 2.1). There exist two types of methods to do so, namely **memory based** and **model based**.

In **memory based** methods, the filtering is made on *neighbours* that can be either items or users. The first case is referred to as *user-based collaborative filtering*, and is the method used in [20]. In order to predict whether an item will be adapted or not to a user, the algorithm relies on the feedback of the nearest neighbours (i.e. the users with the most similar history) of this user on that item. By contrast, in the second case, known as *item-based collaborative filtering*, an algorithm computes the similarity between items based on the feedback they received among users and then recommends an item if it is near enough from those appreciated by a user. One important limitation of memory-based methods is that every user generally explores a tiny zone in the space of items. As a consequence, the algorithm has to deal with sparse feedback data.

Model based methods are more efficient in dealing with these sparse data. To predict an item's interest for a user, they rely on a model (e.g. decision tree, neural network, matrix factorization, etc.) that generalizes on the stored feedback. Recently, they have gained in popularity as they provide accurate results. Although they take advantage of the user community and all the generated feedback, collaborative filtering approaches may struggle when making recommendations for new users or newly introduced items.

This limitation is known as the *cold-start problem*. It is inherent to these methods as they are based on collected data and, thus, require a critical mass of data to be efficient. A way to alleviate this is to enrich the algorithm's representation of users' decision functions, incorporating knowledge of the considered task.

- **Knowledge-based** approaches, as their name suggests, use that idea. These systems can make helpful recommendations even with little or no interaction data by considering how certain features match the user's interests. Therefore, they are mainly used for items with high diversity (and so only a little or no ratings for each particular item) or not frequently purchased, such as real estate or cars. These approaches allow the user to specify what they want explicitly. Then the recommendation process consists in retrieving the items that best match the user-specified requirements. It is where the notion of knowledge comes in: items that match precisely the user demands may not exist, and so domain knowledge is used to compute

similarity metrics. The recommended items are then the most similar to the user specifications. Two main methods exist in knowledge-based recommender systems. On the one hand, *constraint-based* recommender systems ask the user to specify one or more features of interest of the items. For example "I want a single-storey house with a garage". On the other hand, *case-based* methods compute similarity with an example case proposed by the user to choose items to recommend.
The knowledge-based approach is somehow similar to the content-based one as both strongly rely on the items' features or attributes. However, the latter learns from the user's history to make recommendations, whereas the former uses domain knowledge. Generally speaking, the considered knowledge is very user-centric: in constraint-based systems, the constraints or rules are specified by the user, and in case-based, the example is chosen by the user. However, it is noticeable that these frameworks are not limited to user specifications. One can consider systems with other specifications as, for example, not recommending violent movies to young users. This is important as, in a certain way, it meets the perspective of coaching, which considers it profitable for the user to evaluate items not only from his/her view but also with external expertise.

- **Hybrid** recommender systems also exist, which combine two or more of the presented approaches. As we have seen, these approaches are based on diverse inputs that lead to specific advantages and drawbacks. The idea behind hybrid methods is to wisely combine multiple approaches to counterbalance their respective drawbacks and generate better recommendations. The choice of approaches considered and the way their recommendations are combined are the core question of hybrid recommender systems.

2.1.1.3 Recommender systems evaluation

As presented in the previous section, diverse approaches and algorithms exist for RS. Given that, one may compare RS algorithms and settings on a given recommendation task to select the more appropriate candidate RS for this task. This is the problem of RS evaluation. In order to adequately address it and compare several RS, it appears necessary to define appropriate methodologies and criteria. In the following, we present the classical approaches used in the literature.

The first considered question is which methodology to use. As presented in [22], there

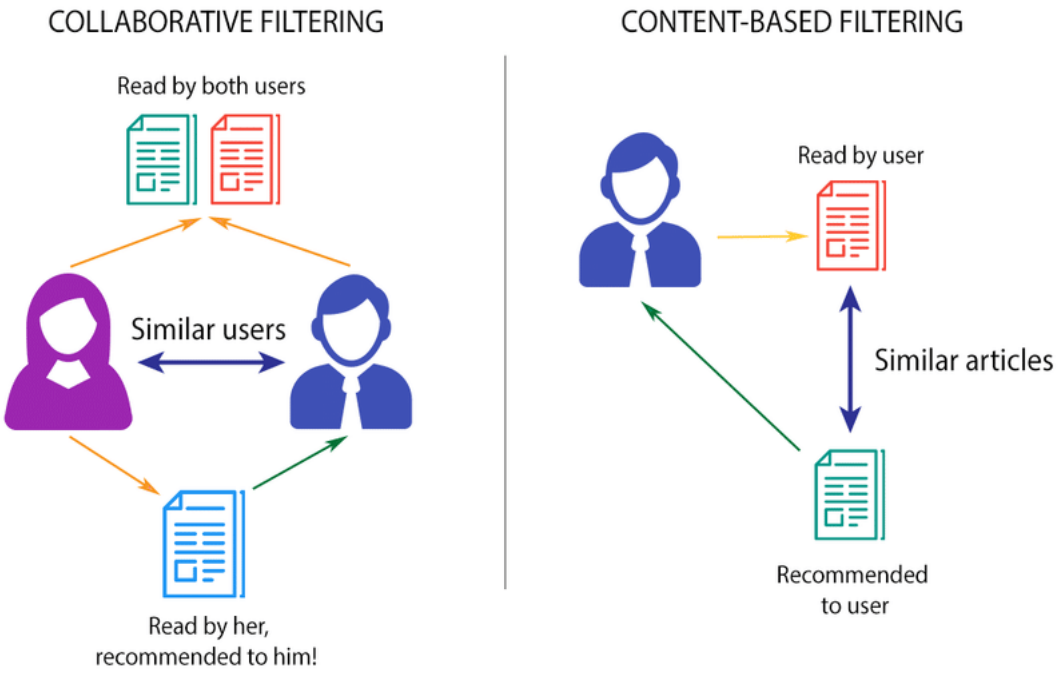

Figure 2.1: Conceptual overview of collaborative filtering versus content-based recommendation [21].

exist three possible methods in order to evaluate RS that range from easy to expensive experimentation:

- **Offline experiments** use a pre-existing data set, gathering interactions between users and items. Then the behaviour of users is simulated, and the performance of the RS is measured on the simulated users. One classical way of simulating user behaviour is to split the data into two sets: one for prediction and the second for validation. The underlying idea is to use the first fraction of data to train the RS as with observed data and then to test its recommendations against the actual following choices of the user. The validation set is considered here as the natural behaviour of the user. The central assumption is that users behave the same when the RS is deployed and at the data collection time. In addition, offline experiments do not consider the possible changes in users' behaviours induced by the RS itself. These limitations make it hard to draw solid conclusions on real-world RS from such experiments. However, offline experiments are very popular in RS literature since they allow for easy implementation and reproducible experiments.

- **User study** is an intermediate approach where a small group of users is asked

to test RS in a controlled environment. Their behaviour is recorded during the experiment, and supplementary investigations can be made, for example, by using questionnaires. Such a method is helpful in that it allows one to gather both quantitative and qualitative data. Since users interact with the system, it also makes it possible to assess the impact of the recommendation on user behaviour, unlike offline experiments. However, user studies also have important limitations. First, they may suffer from various biases, such as user sampling bias or behavioural bias. Second, user studies can be costly to conduct as they are time-consuming for the users.

- **Online experiments** finally are conducted in a real-world setting, that is, when the RS is currently deployed on a platform. The idea is to measure the impact of the RS on users directly. This method is the one that allows one to draw the most robust conclusions, particularly when testing the user behaviour change induced by the RS. For this reason, it is employed in many real-world systems [23]. It is noteworthy that employing such a method may generate a negative user experience if the tested algorithm performs poorly. Thus, in the testing procedure of an RS algorithm, online experiments should be used last, after a strong evaluation protocol has shown shreds of evidence of tested algorithm validity. It is worth noticing that, in essence, online experiments are only applicable on deployed systems, and so mainly on established platforms.

Regarding a recommendation task and its specificity, a wide variety of criteria may be used to evaluate a RS algorithm. Most of the time, several of them are combined and traded off to calculate the overall RS performance. We present the most generally used ones and those of particular interest for this thesis. An extensive list can be found in [22]:

- **Accuracy** is the most commonly used and discussed criterion in the literature, as many recommendation tasks can be seen as prediction tasks (i.e. predicting the rating given by a user to an item or the probability of usage). A classical assumption is that the more accurate the RS, the better the user satisfaction. In addition, accuracy is easy to compute in an offline evaluation scenario.

- **Coverage** represents the proportion of the item set and/or user set that is involved in recommendation. In many applications, especially when considering collaborative filtering algorithms, the RS may focus on a small part of the item set for the

recommendation. In contrast, recommending a wider item selection is often considered desirable. User coverage represents the proportion of users for whom the RS is able to make recommendations.

- **Confidence** is the trust of the RS, whether in its prediction or in the quality of the recommendation. As such, it is an inherent system property and can be computed regardless of the evaluation methodology. It is interesting for the user as it gives him/her a hint about the quality of the recommendation.

- **Novelty** accounts for the capacity of the RS to recommend items not seen before by the user. In many applications, novelty is a key criterion, as known items are much less desirable than new ones for the user. [24]

- **Serendipity** is an evaluation of how surprising the recommendation is for a user. Indeed, in many cases, the recommendation is not expected to be an obvious one for the user. The RS should recommend different items than the ones user would have chosen by himself/herself. In a sense, serendipity can be seen as "anti-personalization": the RS seeks a recommendation that a given user does not expect. Accordingly, serendipity has to be carefully balanced with accuracy to maximise the recommendation's quality.

- **Diversity**, often considered as the opposite of item similarity [25], is hard to define, as it is perceived differently depending on the user. Nevertheless, it has been proven to affect user satisfaction positively [26]. Given this, RS should be able to make recommendations that include items different from each other.

- **Utility** is a general notion that can measure the global interest of the recommendation for the user, the RS or both. Many of the criteria presented above, or their combinations, can be seen as utility functions.

  However, here, what we refer to more specifically as utility is the *value* of a recommendation for the user, the RS or both regarding a defined objective. For example, utility could be the benefit made by the RS owner when recommending items.

  This approach is of particular interest for this thesis work, as we are willing to make recommendations to promote long-term behaviour change. Consequently, the evaluation criteria of the system should encompass not only user satisfaction but

also a measure of behaviour change. Utility appears to be an important notion to evaluate recommendations regarding a combination of these objectives.

More details about the utility evaluation of RS can be found in 2.1.2.3 as utility evaluation is the most commonly used evaluation criterion in the multi-stakeholder perspective of recommendation.

2.1.2. Multi stakeholder perspective on recommendation

As shown in 2.1.1, the framework of recommender systems is pretty flexible as it can deal with a wide variety of items, users and recommendation situations. However, it is noteworthy that all presented approaches are definitely centred on the user. The focus is mainly, if not exclusively, on user satisfaction and how the recommended items meet the user's will. Yet, it may exist situations where other interests could be considered. For example, in an e-commerce application, the item provider may want to sell off his stock or maximize his profit through recommendations made to the customers [27]. In this case, the RS is facing several constraints from different sources. On one side, the item provider is willing to see the most profitable items recommended, while on the other, user satisfaction is still crucial. This is typically the kind of problem addressed by multi-stakeholder recommendation literature. This framework makes it possible to consider, alongside the user interest, other interests such as item providers or system owner ones [28]. This approach to the recommendation problem meets the interests of this PhD work. Indeed, our objective is to design recommendations that promote long-term behaviour change of the user. The multi-stakeholder formalism allows the incorporation of different objectives to the recommendations other than simple user satisfaction. So it appears that it could be a solution to make recommendations that take into account the notion of target behaviour for the user, hence our interest in the corresponding framework.

2.1.2.1 Formal definition

By contrast with the classical approach of RS focused on user interest, the multi-stakeholder perspective considers not only the system and the user as agents but a number of stakeholders who may have diverse interests in the recommendation process. As defined in [29], a stakeholder in this perspective is *any group or individual that can affect, or is affected by, the delivery of recommendations to users.* In the literature, the most commonly

considered groups are the following :

1. **Users**: As in the classical RS framework, users are a crucial point in multi-stakeholder approaches as they are the group that will receive the recommendation. Their interests in the recommendation are the same as in classical RS: they seek valuable recommendations to avoid information overload.

2. **Providers**: They represent agents that furnish the items that are recommended. Their interests in the recommendation can be diverse, but the most common one is to have the items they provide recommended to as many users as possible.

3. **System**: The system is the platform emitting the recommendation. It can be considered at diverse levels, from the company to the algorithm, and the related interests may differ.

Though these groups are heterogeneous and may have different or competing interests, they represent the three main points of view of RS stakeholders. The multi-stakeholder perspective, then, is to consider the (possibly divergent) interests of these groups in the recommendation process. As multi-stakeholder recommendation is an angle given to the recommendation problem, it can be integrated at different levels in the recommendation process. First, one can consider that the performance of a recommendation algorithm has to take into account the diverse stakeholders involved. Doing so may lead to changes in the evaluation function or the measured metrics. Second, the algorithm itself may consider other data sources and objectives to reach the expected satisfaction of the diverse stakeholders. At a broader level, the global design of the recommender system may involve different stakeholders. For example, users and providers may participate in the platform's design.

2.1.2.2 Methods and algorithms

In their general form, multi-stakeholder recommendation systems (MSRS) rely on a notion of utility function when making the recommendation. Indeed, the notion of utility is, by essence, compelling to represent numerous interests that can be opposite. The objective of the RS, then, is not to predict the rating of a given item by a given user but to select the recommendations that maximize the considered utility function. This utility is referred to as an *aggregated utility*, as it combines the aims and objectives of different stakeholders: it represents the global interest of a given recommendation when considering the diversity

of goals. It is essential to notice, though, that the design of the utility function is a key step, as depending on the considered stakeholders and their aims, it can be difficult, if not impossible, to aggregate them via a utility function.

Given this, two main approaches exist in multi-stakeholder literature to maximize utility. The recommendation problem is seen as an optimization problem in the first approach. Then the objective of the recommendation algorithm is to directly maximize the defined utility. For example, in [30] the authors propose to consider the recommendation problem as a *maximum k-coverage* problem and use a greedy approximation to find the best recommendation strategy. Similarly, in [31] a method is presented to optimize the revenue of the item provider at a given time horizon. On the other hand, the second predominant approach in the literature can be seen as successive filters on the items. The assumption is to consider the requirements of each stakeholder separately. For example, the first filter will account for the user requirements (using classical RS algorithms maximizing the considered criteria). On this set of filtered items, a second filter will be applied, accounting for the item provider or system requirements. This approach can be found in [28], where the first generated list of recommended items is filtered to contain at least one item from each item provider, or in [32] where items are re-ranked given a trade-off between fairness and personalization.

2.1.2.3 Evaluation of multi stakeholder recommendation systems

As stated in [29], the evaluation of multi-stakeholder RS is still an open problem as there does not exist a general evaluation framework or benchmark to compare the performance of multi-stakeholder recommendations efficiently. In most cases, the evaluation is made on simulated data, as real data on the system or item providers may be either difficult to obtain or sensitive. As presented above, multi-stakeholder RS use the notion of utility to compute the interests of the different stakeholders and find the best recommendation given these. Thus, in a pretty straightforward manner, the criterion used for evaluation is the evolution of the utility value.

2.1.3. Coaching as a recommendation task

2.1.3.1 Motivation

Given the presented literature and principles, one can consider the coaching problem as a recommendation problem. Indeed, when facing a learner, a coach will recommend him/her actions to take in order to reach his/her goal. As each learner is different, the coach may need to personalize the recommendations in order to find ones that are acceptable for him/her. Considering this framework, the coach can be seen as the **RS**, the learner as the **user**, and the recommendable actions as the **items**. However, one particularity of coaching is that even if the learner knows his/her final objective, he/she is not necessarily able to wisely judge the actions taken regarding this objective and still has his/her own preferences and habits. Thus, the user preferences can conflict with the final objective, given that the coach here faces a trade-off between the user's short-term appreciation of the recommendation and its impact on the user's long-term goal. In other terms, the recommendation is expected to balance the user's perceived utility and the coaching task utility, and so every possible recommendation can be associated with a coaching value or aggregated utility, representing the interest of the recommendation regarding the coaching task. This type of problem is addressed in the multi-stakeholder recommendation literature by approaches that can be grouped under the name of *value-aware recommender system*. Thus, we propose a literature review of the state-of-the-art solutions for value-aware recommendations in the following.

2.1.3.2 Value-aware recommendation: literature review

Value-aware recommendation is a specific problem of the multi-stakeholder literature that emerged mainly in the community of RS for e-commerce purposes. Indeed, much academic research on RS is focused on the user side, trying to maximise the user benefit. However, even if such designed RS are known to positively affect business [33], retailers may want to use recommendation to optimise their profit. Starting from this point, many works were interested in incorporating value (e.g. business value) in the recommendation process. This is the starting point of value-aware RS, where the objective is to balance the interest of the recommendation from the user side and the recommended item *value* (regarding a given value function). Here, we investigate the main papers found on this topic from 2005 to 2022 and propose a review focusing on the approach and the considered

recommendation scenario.

As stated in 2.1.2, there are two main approaches in multi-stakeholder RS: either joint optimisation of both consumer and business value or successive filters (or re-ranking). Table 2.1 shows that both approaches are used in value-aware RS. However, the majority of the investigated papers rely on joint optimisation. One reason for this is the fact that the used optimisation methods allow easy tuning, making it possible to finely balance the weight of user value and business value in the recommendation, as in [34], [35], or [36]. Indeed, according to [37], a trade-off between recommendation accuracy and profitability may exist. Nevertheless, online studies presented in [38], and [34] do not demonstrate such a trade-off and present their methods as increasing profit without significant loss in user satisfaction.

It is noticeable that the vast majority of works focus only on users and system, where the system is also the item provider. In the same way, the considered objectives are almost exclusively business value and user value. However, other parameters are sometimes considered, like recommendation time in [31] or [39]. Indeed, for the latter, the problem is not only finding the optimal recommendation for a given user but also the optimal moment for the system to make the recommendation. However, if the business value considered is nearly exclusively the expected profit (except for [40] and [41] that consider cumulative profit and [35] that considers a general value function), diverse user values are considered, as presented in table 2.2. Meanwhile, if classical user values are sometimes considered [34], the most common way to evaluate user value is to estimate a purchase probability or acceptability of the recommendation. If not the most commonly used method in traditional RS, it is particularly relevant in value-aware RS as it allows a straightforward calculation of expected value.

Regarding the horizon of the value maximisation problem, most existing works focus on the short term. Even if considering a balance between value and recommendation quality for the user, with the intuition that recommendation solely focused on value may be of poor quality for the user and result in poor future outcomes, they do not explicitly handle the problem of long-term effects of recommendation. On the other hand, methods presented in [42], [43] or [44], by explicitly considering a notion of trust of the user in the RS, account for the long-term effects of recommendations solely focused on value. The trust, in these works, represents the user's adherence to the proposed recommendation, i.e. to what extent he/she will accept it. In [43] the system considers the user's trust as an indicator of whether it should focus on value. If the user is in a state where his/her trust

in the system is low (and so is its acceptability of recommendations), the latter will focus its recommendations on the user's appreciation of the items to restore his/her trust. But if the user trust is high, the system takes advantage of it by recommending high-value items that have then a higher probability of being accepted. Another considered long-term effect is discussed in [40], where the authors consider that awkward recommendations can lead to a loss of users for the system. Another approach is proposed in [45]: the authors model the recommendation problem as a sequential decision process, more precisely as a Markov Decision Process. This allows the authors to compute not only the instantaneous expected profit but also the interest of a recommendation in terms of profit when integrated into a sequence of recommendations. So they capture the interest of the recommendation over the length of the considered sequence. However, they do not consider the evolution of the user model over time and the impact on users' habits.

Finally, when considering the evaluation of value-aware RS, one can note that most works focus on offline studies or studies on synthetic data (67% of reviewed papers). The latter generally relies on real data sets, where the business value is not specified, thus generated for the experiment. This can be explained by the fact that business data such as profits and margins are pretty sensitive data for companies and, as a result, are not easily accessible for research. However, many offline studies are also available with real business data. But studies with real users involved in the recommendation process are pretty rare. As presented in table 2.1, only 5 of the 28 investigated studies present methods that were tested against real users. The main reason for this is the expected trade-off between business value and user satisfaction. Indeed, if a terrible recommendation algorithm is not a big problem in offline studies, it is much more so in a real-world scenario with industry constraints. As a result of the dominance of offline studies, most of the evaluation is made on accuracy and observed profit, two metrics that are possible to compute in such a setting.

2.1.3.3 Conclusion

Given the existing literature in the field of value-aware RS, it appears that the coaching problem could be considered a problem of value-aware RS. Indeed, if one can assign a value regarding the coaching goal to recommendable actions, the decision problem of the coach can be seen as a recommendation problem. Following the most commonly proposed approaches, we could consider the user value as the acceptability of the recommendation. Then the value-aware recommendation framework allows for effective consideration of

Recommendation approach	Evaluation method	Horizon	References
Re-ranking	synthetic data	short term	[37], [46]
	synthetic data	short term (tunable)	[47]
	offline study	short term	[48], [30], [35], [39], [49]
Optimization	offline study	short term	[50], [51], [52], [36], [34], [38], [53], [54], [31]
	synthetic data	short term	[55]
	synthetic data	long term	[40]
	multi agent simulation	long term	[41]
	user study	short term	[56]
	formal study	long term	[44], [42]
	formal study	short term	[57]
	online study	short term	[34], [58], [38]
	online study	long term	[43] [45]

Table 2.1: Recommendation approach, evaluation method and evaluation horizon of investigated papers. Synthetic data stand for offline studies with generated value data.

Association rules utility	Click trough rate	Estimated rating	Trust	Content similarity	Purchase probability (acceptability)
[39], [54], [58], [51]	[54], [34],[38]	[49], [35], [46]	[43], [44], [42]	[30]	[43], [44], [42], [48] [56], [41], [40], [36], [55], [50], [45]

Table 2.2: User value considered in the investigated papers

the existing trade-off between the short-term acceptability of recommendations and their impact on the long-term objective of behaviour change.
This framework also allows for a high level of personalisation, which appears to be crucial in coaching. However, there are some key points of the coaching interaction that are not addressed by the value-aware RS formalism. First, state-of-the-art solutions mainly consider short-term goals. Some works integrate an evaluation in the long term, but the only lever considered in the literature is trust. None of the reviewed papers integrates any notion of behaviour change nor the impact of the recommendation on the user's habits. Thus, the considered evaluation criteria do not provide information about the effects after completing the recommendation process.
Second, the literature mainly focuses on value in terms of instantaneous profit for e-commerce companies and does not address the problem of repeated interaction between the user and the system. Most existing work focuses on optimising the recommendation at each step but does not consider the temporal dynamics governing the evolution of the recommendation set. Two noticeable exceptions can be found. In [47], where the authors demonstrated that different recommendations in the early stage of the interaction could lead to different outcomes, with some being more desirable than others. In [45] the authors propose a sequential approach to the value-aware recommendation problem, and so consider the impact of each recommendation on the total profit gain over a limited recommendation sequence.
To conclude, if it is undeniable that coaching problems are in some way recommendation problems, some important particularities remain ignored in the RS literature.

2.2. Inter agents transfer learning

Reinforcement learning [59], known as RL, is a paradigm of learning theory where an *agent* interacts with an *environment* and learns from these interactions a so-called *policy* that determines how the agent interacts with its environment. This framework allows the agent to learn efficient behaviours, particularly in executing sequential-decision tasks. RL is a robust framework as it allows agents to learn how to solve very complex tasks, but it can turn out to be very long for the agent to learn an interesting policy.
Given this, much work has focused on how to reuse this laboriously acquired knowledge and leverage pre-learned policies on related problems to speed up learning. All of these belong to the so-called field of transfer learning. Indeed, as defined in [60], transfer learning

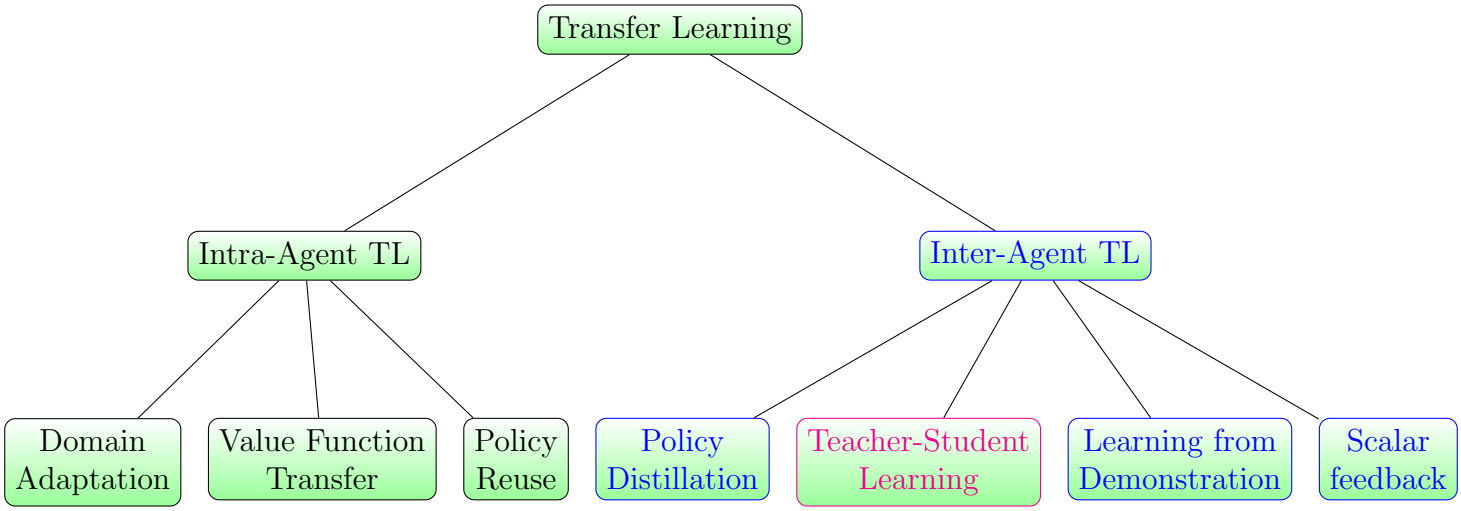

Figure 2.2: Transfer Learning domain description. Here we focus on transfer from one agent to another i.e. Inter-Agent Transfer Learning, and more particularly on teacher-student learning and policy distillation approaches

approaches allow one to use knowledge from one or more *source task(s)* to learn one or more *target task(s)* faster. Transfer learning is investigated in many machine learning fields [61]. However, it is of particular interest in reinforcement learning approaches [62], as reinforcement learning needs a lot of training to converge to the optimal policy. As shown in figure 2.2, transfer learning can be categorized into several subdomains: intra-agent transfer learning and inter-agent transfer learning. In the former, the objective is for an agent to use its own previously acquired knowledge in order to learn more efficiently about the target task(s). On the other hand, the latter focuses on transferring knowledge acquired by an agent to another, avoiding *tabula rasa* learning. Here we are interested in this type of transfer learning as, in essence, coaching involves a notion of knowledge transfer from a seasoned agent (the coach) to a less experienced agent (the learner). In the following, we will first present a quick background on reinforcement learning in 2.2.1, then a general overview of the field of inter-agent transfer learning in 2.2.2. Thereafter we will focus on teacher-student learning and present the corresponding framework in 2.2.3. Finally, we will show to what extent the coaching task can be considered a transfer learning task in 2.2.4.

2.2.1. Reinforcement learning

Reinforcement learning is a learning framework where an *agent* is placed in an *environment*, and learns by trial-error a *policy* that determines how it interacts with that environment. At each step, the agent observes the state $s \in \mathcal{S}$, with $\mathcal{S}$ the set of possible states. Then it chooses to perform an action $a \in \mathcal{A}(s)$ with $\mathcal{A}(s)$ the set of possible actions when in state s. After performing the action, the agent transitions to its new state $s' \in \mathcal{S}$ according to a so-called *transition function* $\mathcal{T}$ so that $\mathcal{T}(s, a) = s'$ with probability $P(s'|s, a)$. Moreover, the agent receives a *reward* $r \in \mathbb{R}$ from the environment. The goal of a reinforcement learning agent is to maximize its cumulative reward all along the interaction. To do so, it has to learn a policy that at each state $s \in \mathcal{S}$ associates an action $a \in \mathcal{A}$, which maximizes the expected return: $\mathbb{G}$. Return is a discounted function of the obtained rewards:

$$G_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + ... = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$$

where $0 \leq \gamma \leq 1$ is a parameter called discount rate. It controls the importance given by the agent to future rewards. Lower values of γ favour short-term rewards over long-term ones, while greater values of γ grants more importance to long-term rewards. Regarding the expected return, the policy π of an RL agent can be represented through an action-value function $Q_\pi : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$, where $Q_\pi(s, a)$ is the expected return when choosing action a in state s and then following policy π. A policy which maximizes the value function for each state s is called an optimal policy and is denoted $\pi^\star$. There exist several algorithms to estimate the action-value function so as to find an estimate of the optimal policy. They can be categorized as *on-policy* or *off-policy*. The former is based on improving the policy used to generate the training data. The most popular algorithm is called SARSA [63]. The latter, by contrast, learn to improve an *action-policy* from data that can be generated by any policy. The best-known algorithm is Q-learning [64] and has been proven to converge to the optimal policy under the assumption that the state space is finite and each state is visited infinitely often.

2.2.2. General overview

As presented, inter-agent transfer learning is focused on transferring knowledge on a given task from an experienced agent to a naive one in order to accelerate the learning of the latter. Thus, inter-agent transfer learning requires at least two different agents that are able to share information: the experienced agent should share with the learner agent so-called *instructions* that aim to accelerate the latter learning on the considered task. As stated in [65], instructions must be specialized to the considered task, available during training and not consider a detailed knowledge of the internal learning parameters. In addition, the user has to be able to interpret and assimilate the instructions.
This framework is relatively general and this allows one to consider many diverse scenarios. It is noticeable that very little assumption is made about the experienced agent and the learner agent. They have to be able to work on the same task, but the fact that in the majority of approaches, the experienced is assumed not to have access to the learner's internal representation allows the agents to be significantly different. For example, one can consider either an artificial agent or a human agent, both as a learner agent and an experienced agent. Regarding the framework, the experienced and learner agent do not even require to share the same action space or observation space: as long as the learner agent can interpret the instruction given by the experienced agent, inter-agent transfer learning is possible. Considering these particularities, it appears that the idea of coaching can meet the principles of teacher-student learning: an experienced coach formulates instruction so that its student learn a valuable policy.
The main question, then, is how the two agents communicate with each other. Classical methods and the associated instruction types are discussed below.

2.2.2.1 Classical methods

Diverse methods have been experienced in transferring knowledge from one agent to another. Here are presented the main methods found in the literature :

- **Learning from demonstration** is a method where the teacher agent presents to the learner agent sequences of states and associated decisions in order to demonstrate how to behave in the presented states. By observing an efficient policy and the associated effects (i.e. state transition, reward, etc.), the learner can update his/her policy and take advantage of the knowledge of the experienced agent to accelerate

learning. Thus, the learner agent does not need to explore the space of all possible actions randomly but can focus on actions that are known to be efficient. In general, each presented sequence is a full episode from the beginning of the task to its end. This method has proven to be efficient on reinforcement learning agents [66]. However, this approach still suffers from certain limitations. First, it relies on an expert on a short predefined period of time [65] and, therefore, is not designed for interactivity with the student and personalization of the advice. Second, the question of generalization from the student given a demonstration is still to be explored [67]. It is also noticeable that both agents need to share the same action space and observation space for learning from demonstration to be efficient.

- **Scalar feedback** [68] is a method where the experienced agent, at each time, can express its satisfaction regarding the learner behaviour, resulting in a *scalar value* communicated to the student. Doing so makes it possible for the learner agent to understand if his/her behaviour is correct and possibly how far it is from the optimal one. In this setting, the experienced agent can interact efficiently with the learner even if the action space and/or the observation space of the two agents are different. Indeed, the scalar feedback is independent of both the action space and the observation space. Although interesting, this method has proven its efficiency only with human teaching artificial agents. Indeed, the scalar feedback in itself is particularly well adapted to teach artificial agents as it can easily be directly assimilated to a reward function. In this sense, scalar feedback can be seen as some sort of reward shaping [69] [70]. In addition, scalar feedback is not well adapted to generate personalized instructions. In this setting, the experienced agent evaluates the learner's behaviour in light of its own policy. However, depending on the learner agent's specificity, the best policy for the learner may not necessarily be the same as the experienced agent's. That makes it difficult for the experienced agent to adapt its feedback to the specificity of the learner agent.

- **Teacher student learning** is a method relying on action advice from the experienced agent to the learner. In this scenario, the experienced agent acts as a teacher by providing *suggestions of actions* to the learner that are interesting in the considered state. In this setting, the teacher and the learner only need a joint action set to be applicable [71]. Their decision functions and representation of states may be completely different. In addition, as the advice is given all along the interaction,

the teacher can focus on particular subsets of the state-action space that are critical for learning.

- **Policy distillation**, although presented here along with the three other methods, proposes a slightly different conception of the problem. Policy distillation stems from the idea of *knowledge distillation* [72], which is a form of model compression [73]. In this approach, an agent's policy is assimilated to a prediction model that associates with each encountered state an action. Indeed, the knowledge distillation principle originally came from the supervised learning literature. In policy distillation, the student agent is assumed to be an untrained model taught by a trained model, i.e. an experienced agent. The core idea of policy distillation is for the student agent to use supervised regression to train its internal model (defining its policy) so as to produce the same output distribution as the model of the experienced agent. This approach has proven to be efficient on diverse tasks, such as transferring knowledge on how to play Atari games [74]. Moreover, the distilled policy can lead to better performance than the teacher. According to [74], crucial attention must be paid to the regression loss function when considering policy distillation approaches.

  Although an actual method to manage knowledge transfer from an experienced agent to a naive agent, the policy distillation framework is definitely centred on the student. State-of-the-art solutions focus on how for the student to learn efficiently from the trained model outputs. As such, policy distillation is closer to the *imitation learning* problem, where an agent tries to learn a policy from observing an oracle. As our work focuses on designing recommendations, we are more interested in methods relying on the experienced agent rather than methods for the student to maximize his/her learning.

In this thesis work, we investigate the coaching problem, where a personal automated coach repeatedly advises a user to accompany the latter in a behaviour change process. Regarding this, work on teacher-student learning is particularly interesting for our problem. Indeed, teacher-student learning fits the imperatives of coaching, as it allows the teacher and the student to have different knowledge representations and rely on action advice which supports personalisation. In addition, one can find in the literature works with human students [75]. The developed idea of artificial agents teaching humans a specific task definitely meets the concept of coaching. More generally, the study of how to teach and how to learn how to teach is of great interest to this thesis work. Thus,

in the following, we present the framework of teacher-student learning; we examine the state-of-the-art solutions and discuss their relations to the coaching problem.

2.2.3. Teacher student learning

Works on the teacher-student framework, as defined in [71], are interested in the problem of finding the most efficient way for an experienced agent (i.e. the teacher) to interact wisely with a naive one (i.e. the student) in order to accelerate the learning of the student by transferring him/her knowledge previously acquired by the teacher agent. As we have seen, teacher-student learning relies on a type of interaction called *action advice*. In this setting, the objective of the teacher agent is to suggest to the student actions that will significantly affect their learning.

2.2.3.1 Definition of the framework

In general, the literature on teacher-student learning considers two different scenarios guiding the interaction between the teacher agent and the student agent: the *learner-driven* and the *teacher-driven* scenario. In the former, the learner engages in interaction with the teacher. So the learner first interacts with the environment independently and thus needs a proper policy. At any time, he/she may ask the teacher for advice. The teacher can then decide whether or not to give feedback regarding several parameters such as the possible number of interactions with the learner, the interest for the learner of the asked advice, or its own confidence in the given advice. When the advice is given, the learner applies it and updates his/her policy.
Conversely, in the latter scenario, the problem of when to advise the learner is handled by the teacher agent. Then the teacher agent has to evaluate the need for advice from its interactions with the learner and its efficiency in the task. These two scenarios seem similar when considering the high-level task of learning an effective policy. However, this slight change leads to significant differences in the approach for the teacher agent. Indeed, being able to decide when to advise the student needs for the teacher to monitor his/her behaviour efficiently and to identify when given advice will have a real effect on student learning. In a learner-driven setting, the teacher agent is solely focused on the problem of *which* advice to give. Conversely, in the teacher-driven setting, the teacher agent has to answer both the question of the advice production and the question of the advice distribution (i.e. *when* to advise the student agent).

Thus, one can consider two distinct sub-tasks: advice production and distribution.

The production task covers the question of *which* piece of advice to give to maximize student learning. In the literature, the advice production issue is generally handled by exploiting a learned policy for the task to be taught. In this setting, the teacher has explored the state action space and so is able to suggest the best action to the user regarding its policy. However, it is noticeable that the performance of the teacher on the considered task does not guarantee its performance as a teacher. For example, [76] shows that the best teacher is not necessarily the one who performs the best at the considered task.

Besides the advice production task, the distribution task solely focuses on *when* the student should be advised. As we have seen, this task can be handled either by the student or by the teacher. The approaches where the student decides when to be advised often rely on a more or less complicated notion of confidence in its own policy. For example, the student in [77] estimates the novelty of pieces of advice, while [78] lets the student calculate the criticality of its state to determine if it needs advice or not. On the other hand, when relying on the teacher side, the advice distribution issue is either handled by human-shaped heuristics [71] [75] or by learning [76] [79]. A particular heuristic that requires two-step communication between the teacher and the student is the so-called *mistake correcting* [71]. In this setting, the student announces to the teacher his/her planned action, given the state it observes. Then the teacher can determine if the action is good or not, that is, if the student makes a mistake. If so, it can decide to provide advice, correcting the student's mistake.

A related problem to the advice distribution is the advice budget. Indeed, if the teacher is able to advise the student all throughout the interaction, one can consider that the teacher should always give advice. However, communication between agents is, in many real-world problems, a scarce resource. Moreover, it seems unrealistic to consider an unlimited advice budget when considering human agents with limited attention. Thus a vast majority of the literature on teacher-student learning considers a limited budget of advice for the teacher agent. The question of when to advise then becomes the question of how to make every single piece of advice the more impacting possible for the student's learning.

2.2.3.2 Evaluation

The objective of teacher-student learning is to accelerate the learning of the student agent on a given task. Therefore the evaluation of a given method should be examined regarding the time the student takes to master the considered task. However, such an evaluation is not straightforward, as diverse aspects of time performance may be considered. As a transfer learning task, teacher-student learning can be evaluated by using transfer learning evaluation methods. As presented in [62], the evaluation of transfer learning is complex, as there does not exist a single metric encapsulating all the dimensions of performance. Instead, the authors propose to evaluate transfer learning in a multi-dimensional fashion, considering diverse metrics. Each of these metrics captures a different aspect of performance. The metrics proposed by the authors are the following:

- Jumpstart: This metric accounts for the difference in the initial performance on the considered task of an agent using transferred knowledge when compared with the performance of the agent's initial policy. It is particularly relevant when transfer comes from a pre-training phase, like in *learning from demonstration*.

- Asymptotic performance: This metric compares the final performance of the student agent once its learned policy has converged, with and without being taught. This has two major limitations: first, the convergence may be very long to attain, which is particularly true on reinforcement learning tasks with infinite state-action spaces. Second, the asymptotic performance does not take into account the time needed for the student to converge, which is yet critical in many settings.

- Total reward: Comparing the total reward obtained along an episode of the task by a taught agent and a naive one is a common metric. It allows one to measure the total gain induced by the teacher agent, and so is some sort of dual measure, accounting both for the convergence speed and the convergence level of the student. However, as such, it is highly dependent on episode duration, the reason why it should be preferred for tasks performed during a given time.

- Transfer ratio: Transfer ratio is the ratio of the total accumulated reward by the taught student and a naive student. It is particularly adapted when comparing different teaching methods, as, in a way, it accounts for the quantity of knowledge effectively taught to the student. In particular, it allows making a distinction between teaching methods that leads to the same final performance. However, like the

total reward metric, the transfer ratio directly depends on the number of learning steps considered.

- Time to threshold: By measuring the time (i.e. the number of learning steps) to reach a predefined performance threshold, one can characterise the agent's learning speed and so to the acceleration allowed by the teacher's advice. The major drawback of this metric is that it necessitates defining a threshold, which is, in essence, task-dependent.

The presented metrics capture diverse dimensions of teacher-student learning performance. Thus, evaluating different teaching methods and efficiently comparing them needs to consider a combination of these metrics to be complete.
When considering the evaluation of a teaching task, it is also essential to differentiate the two aspects of advising: advice production and advice distribution. Yet, the literature on teacher-student learning does not have elaborated benchmarks to compare these two sub-tasks separately. However, the vast majority of works solely focus on one of the two aspects and then compare different approaches through the metrics presented above.

2.2.4. Coaching as a transfer learning task

2.2.4.1 Motivation

Considering the presented framework, our problem of making recommendations for long-term behaviour change can be modelled as a transfer learning problem and, more specifically, as a teacher-student learning problem. Indeed, we can view the recommender system as a teacher agent that provides advice to the user to facilitate his/her behaviour change towards a goal. Moreover, such a recommender system should be able to repeatedly give his/her human user personalized instruction and adapt it in regard to user specificity.
In this setting, the recommender can be seen as the teacher agent and the user as the student agent. Thus, the recommendations of the system play the role of advice from the teacher. One specificity of our problem lies in the fact that the user is not able to draw a direct link between his/her actions and his/her final objective. This makes it impossible for him/her to evaluate whether or not he/she needs advice. Given that, the problem can be seen as a teacher-driven teacher-student learning problem.
In the following, we present a brief state-of-the-art of teacher-student learning literature and review the main characteristics of the existing methods. Given the nature of the

Focus \ Scenario	Teacher driven	Student driven	Jointly driven
Advice production	[81], [82], [83], [84], [76]	[85]	[86]
Advice distribution	[71], [87], [76], [79], [88],[75]	[78], [89], [90], [91], [77], [85], [92], [93]	[94], [86]

Table 2.3: Characterization of the investigated papers regarding interaction scenario and focus of the study. A jointly driven scenario refers to a student-driven scenario where the teacher can decide whether to give the queried advice or not.

question addressed in this thesis, we mainly focus on works about the teacher-driven scenario. Indeed, we are interested in the question of how to produce and deliver advice, such as to maximize the learning of the student, rather than in the question of how to make the best use of the advice for the student.

2.2.4.2 Review of state-of-the-art solutions

The teacher-student framework was first described in [80] with a focus on student-driven interaction. More recently, [71] considered the problem of teacher-driven teacher-student learning, integrating the notion of interaction budget. This work has raised the community's interest and led to the development of specialized literature on the teacher-student framework by defining a precise yet flexible framework and presenting promising results for reinforcement learning agents. Here we present a review of the significant works in this field from 2013 to 2022.

As presented in table 2.3, the investigated work can be characterized regarding the considered interaction scenario and the focus of the study, whether on *production* or *distribution* of the advice.

First, let us evoke works on the student-driven scenario. As one can notice, except [85], all the investigated works solely focus on the problem of advice distribution. As we have seen, in the teacher-driven scenario, advice distribution is handled by the student: it decides when to ask for advice from the teacher. Therefore the corresponding works mainly investigate the student decision function to maximize learning. Most works rely on metrics that the student computes to characterize the encountered states. It can be state criticality [78], state novelty [77, 92] or student confidence in his/her policy on a given state [93, 85]. Another direction is investigated in [89], [90] or [91]. These works

propose methods that let the student reuse previously given advice. Then the student has to decide in each state if it should ask for advice, reuse advice or follow its policy. To do so, the authors augment the student with additional policies, driving the management of the previously received advice. In this sense, the authors of these works make extra assumptions about the student model.
Conversely, in the teacher-driven setting, the problem of advice distribution falls to the teacher agent. Thus, the solution must rely on something other than a direct quantification of student learning, as the teacher is assumed not to have access to the internal parameters and representation of the student. The corresponding works can inform us on when to make advice to maximize their effect, which is of interest to our recommendation problem. Two main approaches were investigated in the literature.
The first one relies on heuristics that guide the distribution of advice. Different methods were proposed. A widely studied one is the so-called *early advising*: it is based on the intuition that the early stages of learning are critical, and the idea is then for the teacher to use its advice budget in the first interactions with the student. This method is tested in [71] and [87] on diverse reinforcement learning tasks. An improvement of early advising proposed in [88] consists in advising on the early learning steps, but alternatively, every m step. This way, the time the teacher takes to spend its budget can be controlled by the parameter m, and advice can cover a more significant part of the state space. Another commonly used heuristic is known as *importance advising*. In this setting, the teacher agent computes an "importance" metric for each state regarding its knowledge of the task and then focuses its advice on the crucial states. As stated in [71], considering a reinforcement learning agent as the teacher allows to compute the importance metric in a pretty straightforward manner: the authors used the difference between the maximum and the minimum of the state-action value function in state s. However, it appears that considering measures based on variance or absolute deviation of the state-action value function, as proposed in [88], produces more consistent results. This heuristic is tested in [87] on a domestic robot scenario and in [75] on human students learning to play a video game.

When possible, one can also consider *mistake correcting*. Indeed, this method necessitates an additional communication step between the student and the teacher. At each step, the student announces his/her intended action to the teacher. Then if the action is not considered good by the teacher (i.e. the student makes a mistake), the latter makes advice if the advice budget is not exceeded. This method has proven its efficiency against other

heuristic methods [71], [87], [88]. The only investigated study with human students also points to the superiority of *mistake correcting* regarding the other tested advising methods [75]. However, it necessitates extra communication at each step which is often costly. To overcome this limitation, predictive advising was proposed, where the objective is for the teacher to predict the probability of student mistakes and decide whether or not to advise. An implementation using SVMs to predict the student intended action was used in [71] and showed promising results.

The second investigated approach in the literature for the teacher-driven scenario is to model the advice distribution task as a reinforcement learning task. This idea was first introduced in [79]. In this work, the authors propose a reinforcement learning model for the teacher, with a reward based on the student's time (i.e. the number of steps) to reach a given goal state and the reward associated with that goal. They trained their teacher model to teach a student on a classical reinforcement learning benchmark (i.e. mountain car problem [95]) and showed that their reinforcement learning teacher outperforms the best heuristic strategy. Indeed depending on the considered budget, their reinforcement learning teacher leads the student to better or equal performance than mistake correcting. With a different conception of reward, [76] also proposed learning an advice distribution strategy. They introduce *Q-teaching*, a learning-to-teach algorithm based on the Q-learning algorithm, a classical off-policy algorithm for reinforcement learning problems [59]. According to the authors, the key insight of Q-teaching is rewarding the teacher in regard to the value gain it allows for the student. Indeed, the algorithm uses the difference in value between the advised action and the value of the estimated student action as a reward. The authors propose two versions of the algorithm based on how the student action is estimated. The off-student policy Q-teaching uses a pessimistic estimation of the student action. By contrast, the on-student policy Q-teaching uses the actual action of the student. The authors tested both algorithms on the task of teaching reinforcement learning agents how to play the Pac-Man game. The results showed that the proposed algorithms perform similarly but slightly worse than mistake correcting and Zimmer [79] algorithm on this task. However, the authors noticed that off-student policy Q-teaching needs significantly less training than Zimmer's algorithm and does not need to know the intended action of the user to reach similar performance. According to the authors, the results of on-student policy Q-teaching are probably due to the variability induced by the student policy that necessitates a much longer training phase.

As one can notice, most works on teacher-student learning focus on advice distribution and consider only the action learned by the teacher as an advice source. However, producing relevant advice is a challenge in the field and is discussed in several works. The first investigated question is that of choosing the best teacher for a task. In [84], the authors show that the variability of the teacher's behaviour for the task is important when considering its students' performance. They trained reinforcement learning agents on a given task, with two possible paths to the final state, one better than the other in terms of total reward. They show that three major types of agents emerge after training: agents specialized in the best path, agents specialized in the longer path and polymath agents without a clear preference. Then the three types of agents are used as teachers for the task. The authors show that the best-performing teachers are polymath agents, although they are neither the best at completing the task (shortest path agents) nor the most exploratory (longest path agents). They notice that polymath agents are agents for which the reward standard deviation on the task is the lowest. Moreover, they also find that the lower the student obedience (i.e. probability of accepting the advice), the stronger the effect of advice consistency: higher consistencies leading to better performance.

Investigating the advice variability, the work of [76] show similar results by observing better teaching performance for teacher with lower reward coefficient of variation (computed from standard deviation) on the task. In particular, they show that a teacher trained using the R-learning algorithm [96] is better at the teaching task than other Q-learning teachers while being significantly worst at performing the task. Another investigated research direction is how the teacher may learn to give advice. In [86], authors show that using direct student task performance as a reward when learning to give advice can lead to poor learning. Indeed, following conscientiously advice and accumulating reward for the student does not guarantee the actual learning of an interesting policy. Thus, the authors propose to reward the teacher according to the student's progress instead.

In [83], the authors propose a method where the teacher agent advises for sub-goals on the taught task in order to ensure step-by-step learning. They show the efficiency of the method in a cooperative multi-agent reinforcement learning environment against diverse state-of-the-art methods.

Finally, some studies consider the problem of enriching the advice provided to the student. The work of [81] suggests letting the student be capable of building decision trees from the teacher's advice to memorize it and use it even in not yet encountered states.

Moreover, they present a method to extract advice in the form of recommendation trees from the teacher's optimal policy. This method allows the generation of both advice (leaf of the decision tree) and explanation (path in the decision tree). Doing so makes the advice more understandable even for a human being student. The framework is then tested on a multi-agent reinforcement learning environment and shows improvements both in convergence speed and total reward when compared to classical benchmarks. With the same idea of advice enrichment, [82] proposes a method that categorizes the states encountered by the student. By doing so, the teacher agent can give the student hints on the long-term profitability of the visited states via qualitative assessment. They show that their method improves the re-usability of the given advice and outperforms baselines on teaching students to play the Qbert video game.

2.2.4.3 Conclusion

Regarding the teacher-student learning framework and the related literature, one can definitely model the problem of long-term behaviour change recommendation as a teacher-student learning problem. In this sense and contrary to the value-aware RS formalism presented in 2.1.3, the teacher-student learning framework makes it possible to handle long-term goals as the notion of behaviour change is inextricably linked to that of student learning. Moreover, the framework in its definition considers repeated interaction between the teacher and their student. When considering the distribution of advice, it appears that *mistake correcting* is one of the most efficient methods existing in the literature that is adaptable in the context of recommendation. One of the main results from the literature on advice production is that the best teacher is not necessarily the best agent on the task. This has to be taken into account when designing recommendations.
However, the framework still suffers from limitations in modelling our investigated recommendation scenario. First, all presented works assume a reinforcement learning agent as the student that learns on his/her own. This is not necessarily true for the problem addressed in this thesis, as the reward function may not be accessible to the user. Consider, for example, a scenario where a user is coached to drive a car in a more environmentally friendly manner. The exact impact of each recommendation is probably not accessible to the user. Yet he/she does not have direct access to a reward function and thus cannot learn entirely on its own.
Another critical point is the question of advice personalisation. If we have shown that action advice is the more user-adjustable method for transfer learning, the presented

works do not consider the student agents' diversity. In particular, and even more with human students, all the students may not be able to reach the same final performance level, as they may not be able to follow every policy. This problem is not addressed in the investigated literature. The work of [84] considers the notion of obedience, which can be seen as the acceptability of the advice for the student. However, they do not make a difference between suggested actions or between users. And yet this is key in the coaching problem as some actions can be easier than others for a given user.

To conclude, if some of the questions addressed by teacher-student learning are of interest to the coaching problem, one cannot be included in the other. Insights about how to teach and how to learn how to teach are of particular interest to our work. However, the teacher-student framework still suffers from limitations in modelling the coaching problem.

2.3. Technologies for behaviour change

With the emergence of Artificial intelligence, many works focused on designing technologies to teach or help humans change their habits. This interest is not new: as early as 1924, Sidney Pressey conceived a mechanical machine aiming at teaching students without the intervention of an external teacher [97]. But AI methods make it possible to achieve both an enhanced learning interaction and a substantial level of personalization. Using these levers emerged the field of Intelligent Tutoring Systems (ITS) [98], which are able to teach human students in diverse disciplines on their own. Besides that, the pervasiveness of computers and smart devices in our lives led researchers to consider using these to induce behavioural changes in their users. This is the perspective of Persuasive Technologies [99, 100].

With different approaches, these two disciplines use technology and human-computer interaction to induce behaviour change. Either in an informed and development-driven way using actual learning or in an influencing, performance-driven way using persuasion. As such, the approaches and methods developed in these are of interest to this thesis work. Indeed, in a coaching interaction, the learner is informed of his/her final objective but not necessarily of the steps to reach it. In addition, some recommended actions can be counter-intuitive for the learner while leading to gains in the long term. Thus the coaching problem can be considered at the intersection of development-driven and performance-driven approaches [101], and as such at the intersection of persuasion and

learning perspectives. In the following section, we will present the state-of-the-art and classical methods of these two approaches and situate coaching in this body of literature. In Section 2.3.1 we will present an overview of persuasive technologies, their methods, and application cases. In Section 2.3.2 we will present an overview of individual tutoring systems and state-of-the-art solutions. Finally, in Section 2.3.3, we propose to consider coaching as a technology for behaviour change and discuss how the presented literature may inform the design of coaching systems.

2.3.1. Influencing perspective : persuasive technologies

Persuasive technologies, known as PT, can be defined as "any interactive computing system designed to change people's attitude or behaviour", according to Fogg [100]. The key point here is that the purpose of changing users' attitudes and/or behaviour is the one the system is designed for. Indeed, the majority of information systems may have an influence one way or the other on user behaviours, while it is not necessarily their objective as stated in [102]. In this sense, PT is more of a system design philosophy, a general framework, than a precise algorithmic approach. Thus, it can be applied in numerous domains with diverse implementation strategies. Consequently, literature on PT is fragmented [103] and building a clear overview of the field is challenging. This necessitates, in particular, a systematic categorization of the applied methods and concepts. Some literature reviews exist [104, 103, 105, 106, 107, 108] that propose such classification, and we will here present a summary of the main categories presented in the literature.

2.3.1.1 Overview and categorization of persuasive technologies

Given the diversity of the approaches, PT can be categorized according to many different characteristics. Here we will focus on the more prominent ones. First, regarding the actual target of PT, authors of [102] make the difference between three types of changes that could be pursued, ranked by order of difficulty.

The first and easiest step is *change in the act of complying*. Here system's goal is simply to make the user perform an action. It does not necessitate the user to be motivated for it, nor seek to change his/her behaviour in the long term. The pursued change is only an act of instantaneous compliance and, as such, can be defined as "nudging" the user to the objective.

The second level the authors present is *behaviour change*. They define it as a more

enduring change that should be repeated when in the same context.
Finally, the third considered possible change is *attitude change*. Attitude is a more high-level concept, defining how a user will consider his/her environment. An attitude change often leads to several behaviour changes in diverse contexts.
In addition to target changes, the authors also define three possible outcomes of PT :

- **Forming** a behaviour refers to the fact of generating a behaviour responding to a new situation. As such, it encompasses both the identification of the situation and the formation of the corresponding response.
- **Altering** a behaviour is defined as the fact, for the target, to change its behaviour in response to a known situation.
- **Reinforcing** a behaviour applies to situations where an observed behaviour is reinforced to become more persistent and/or frequent.

Although of a high level and not giving direct implementation clues, these two axes of categorization are of great interest when considering the PT literature. Indeed, the design methods can significantly vary among the nine possible drawn objectives, and a clear definition of the latter seems to be a sine qua non to further considerations.
Regarding the actual methods involved in the production of PT intervention, one should consider two essential questions: the choice of the behaviour model on which the system is based and the motivational strategy used by the system to persuade the user.

2.3.1.2 Theoretical behaviour models

Let us first consider the question of the behaviour model. In [103], the authors investigated 85 publications in the field of persuasive technologies applied to the human health domain. They noted that most of the investigated studies do not specify any theoretical framework for behaviour modelling. The remaining studies used many different models, with 21 considered models for a total of 34 studies, some studies relying on several models. The same tendency is observed in [106], where 24 of the 44 reviewed studies do not rely on any theoretical behaviour model and a great diversity of considered models in the remaining studies. The authors of [104] present this lack of a unifying theoretical framework as an important pitfall of the literature. In particular, they emphasise the importance of considering the notion of habits when designing PT for long-term behaviour change. We will not discuss the existing theoretical behaviour models here but rather present the

framework they developed. For a more in-depth presentation of the most used models in PT literature and their limitations, refer to [104].
The framework described in [104], known as the *Habit Alteration Model* is a synthesis of three pre-existing models: Dual Process Theory, modern habit theory, and Goal Setting Theory. It consists of three layers (or phases) of behaviour generation and two types of processing. Type 1 refers to automatic behaviours (e.g. habits), while type 2 refers to conscious behaviours (e.g. intentions). Behaviour is considered a function of a context. The first phase of behaviour generation is the perception of the context, which can be conscious (Type 2) and/or unconscious (Type 1). The two types of perception lead to consider the *cues* that are actually perceived. Then begin phase 2 of behaviour generation: preparation of the action. The detected *cues* generate either one or more *impulse* (Type 1 processing) or one or more *intention* (Type 2 processing), both known as *responses*. These are added to a so-called *potential response stack*, which ranks potential actions regarding several factors: match with the particular *cue*, affect towards the *cue* and/or *response*, and accessibility. Then in the last phase, the acting phase, the higher response on the list is chosen as a behaviour. If there is a tie between an impulse and an intention, the impulse is preferred as it is processed faster by essence. According to the authors, the *Habit Alteration Model* fills the gap between approaches of behaviour solely based on environment and approaches solely based on internal cognitive factors and makes it possible to design PT based on habit formation.

2.3.1.3 Motivational Strategies

One can find a wide variety when considering actual methods used by PT systems to persuade their users. Here we present the most significant ones :

- **System generated feedback** consists in informing the user, after he/she performed an action, on the interest of this action regarding the pursued goal. Feedback can be given in a visual, audio, or textual fashion and is generated by the system (conversely to social feedback). As an example, works on persuasive chatbots often use textual feedback [109].

- **Tracking and monitoring** methods are based on furnishing information about the user's past behaviour. The idea is to give the user factual data about his/her behaviour. Allowing the latter to have an overview of its last actions makes it

possible for him/her to realise his/her potential mistakes. According to [103], it is the most used method in persuasive technology for health purposes.

- **Social interaction** is a vast family of methods based on what is called "social acceptance" in the Fogg behaviour model [110]. It can go from social feedback to comparison to others. The critical dimension in this kind of method is the exchange with other human beings.

- **Persuasive messages and reminders** are used to help a user not to forget an objective and/or to persuade him/her to adopt a specific behaviour in a given context. Although pretty popular, these methods necessitate an effective categorisation of the context, which can be challenging [104]. In addition, these methods induce cognitive load of the user, which can induce an ineffectiveness of the messages.

- **Suggestion and advice** is a method that used to be pretty popular according to [106] and [108]. It consists in making suggestions or recommendations to the user when acting. As such, it matches the focus of this thesis.

2.3.1.4 Conclusion

The literature on persuasive technologies is notably fragmented. The field is more defined by the behaviour change objective rather than a methodological unity. As presented, most PT's underlying behaviour models are limited and do not ensure a solid theoretical validation. However, the field is considered by many researchers as very promising, regarding both the stakes it addresses (e.g. public health [103], environmental sustainability [107]) and the increasing use and pervasiveness of technology in everyday life. Moreover, when investigating the actual effects of PT, one can find encouraging results: the study of 95 publications in the field performed by [105] showed that 54.7% of the studies reported fully positive results. In contrast, 37.9% reported partially positive results, that is, according to the authors, situations wherein some but not all of the studied elements showed positive results. In the specific domain of health and well-being, these results seem even better, with 75% of the 85 investigated studies showing fully positive results and 17% reporting partially positive results. To conclude, we can say that the persuasive design of systems is promising but requires special attention to methodology.

2.3.2. Learning perspective : individual tutoring systems

Individual tutoring systems (ITS), as defined in [111], refers to "the interdisciplinary field that investigates how to devise educational systems that provide instruction tailored to the needs of individual learners, as many good teachers do". It is a sub-field of artificial intelligence focused on educational purposes. One major particularity of ITS is that they are aimed at providing individualized support for learning activities in a one-to-one interaction. As such, it involves two agents interacting with each other. First is the student or tutee. He/she is a human being working on a given *task* and is expected to train to achieve this task better. The second agent, known as the *tutor*, is an automated agent or system aiming to teach the student on the task. As stated in [112] the system is expected to know what it teaches, whom it teaches and how to teach the considered task. In order to fulfil these requirements, the classical architecture of ITS is based on four interacting components [113] :

- A **task environment** or interface, that is used by the tutor to gather information about the way the user performs the task. Collected data can then be processed by the system to deliver instructional actions or feedback through the interface module.

- A **domain knowledge** module. The system uses it to evaluate the actions taken by the student. To do so, it models expert knowledge about the task. In particular, it can generate solutions to the considered task and compare them with student actions.

- A **student model** that allows the system to follow the student's progress and to model his/her current knowledge on the task. The student model is key in ITS as it informs the system on the aspects of the task it should focus its interventions. It is generally composed of an estimation of the student domain expertise based on the system domain knowledge and a catalogue of student misconceptions on the task.

- A **Pedagogical module** aims at determining what type of intervention should be delivered to the student. It may consider two different levels, the direct task level or a higher level of task sequencing in a particular domain. In [114], five instructional actions are considered for the intervention of the system: performance demonstration by the system, Directed step-by-step performance where the student follows the steps given by the system, monitored performance where the system corrects

the student mistakes and goal seeking where the system monitors the abstraction on the task and free exploration of the task by the student.

The strength of the ITS approach mainly lies in the domain knowledge module. Indeed, cognitive science and psychology are used to model tasks using human-understandable representations. This makes it possible for the system to reason on the task via human-achievable actions. By doing so, the system can model student abilities and aptitudes and provide personalized instructions. Although an interesting approach, individual tutoring systems are definitely focused on academic tasks and educational purposes.

2.3.3. Coaching: a technology for behaviour change

As stated in the introduction of this section, the automated coaching problem can be considered at the intersection of the learning and influencing perspectives, and so lies in the field of technologies for behaviour change. As such, the design of an automated coaching system has to consider guidelines emerging from the field and, in particular, a valid user behaviour model, which appears to be a key determinant of persuasive technologies and intelligent tutoring systems. The student models developed in the ITS field can be considered. However, they mainly focus on a very scholarly approach to learning, which is not the best fitted to everyday behaviour change targeted by coaching. On the other hand, as we have seen, the persuasive technology literature proposes a tremendous number of behaviour models with numerous pitfalls. However, the *habit Alteration Model* described in [104] seems to overcome many of these and is thought to form long-term user habits. This makes it particularly relevant regarding the problem of coaching. Thus it seems relevant to model the coaching problem as a persuasive technology problem with a focus on habit forming.

Even though important in guiding the design of systems and particularly of the user interfaces, we did not find in the persuasive technology literature precise lower-level methods that can be applied to the problem of recommendation production. Therefore, modelling the coaching problem as a persuasive technology problem shows some limitations in directly guiding conception and implementation.

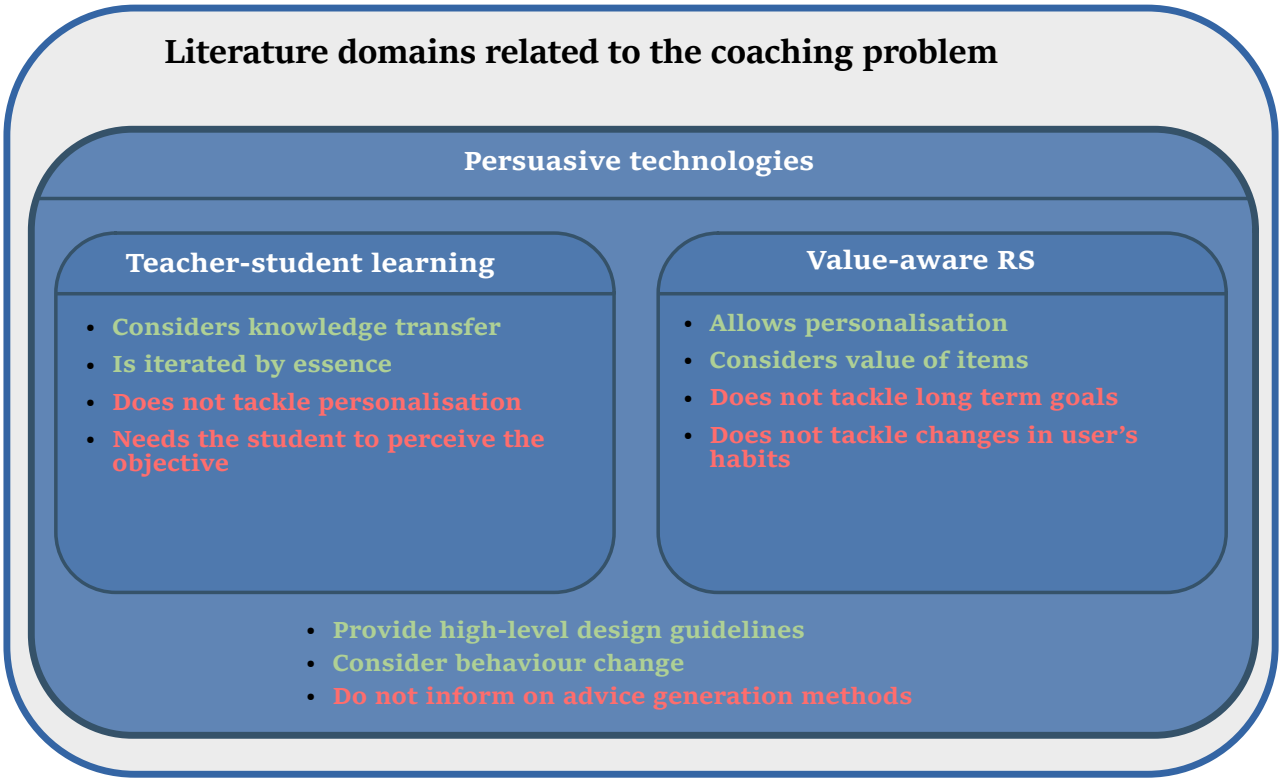

Figure 2.3: Presentation of the reviewed literature contributions and pitfalls regarding the coaching problem

2.4. Chapter Conclusion

In this chapter, we proposed to explore different dimensions of the literature when considering strategies to induce lasting changes in behaviour at a user level. We first considered a recommendation systems perspective, showing that the coaching problem can be modelled as a multi-stakeholder recommendation problem. However, we underlined that such a framework fails to consider both the long-term effects of the recommendations and the iterated nature of coaching, with an important repetition of interactions.
We then presented the coaching problem as a transfer learning problem, modelling it through the teacher-student framework. But such a model, if useful to consider both the knowledge transfer from an experienced agent to a naive one and long-term dynamics, still suffers from limitations when considering a coaching interaction. In particular, it appears

to be limited in modelling users pursuing goals that do not lead to a direct reward. In addition, the reviewed literature shows a general lack of effort in personalization, which is a key component of coaching.

Finally, we explored the literature related to technologies designed for behaviour change and showed the coaching problem can be modelled as a persuasive technology problem. But if such a model draws a general design pattern for persuasiveness, it does not inform us on the actual technical solutions that can be conceived.

To conclude, we can see that the literature, while investigating related questions, does not explicitly address the problem of coaching as defined in 1.3. From this statement emerges the importance of defining a new framework to encompass all the dimensions of a coaching problem and to study it formally. Nevertheless, it is noticeable that the presented literature, by capturing diverse dimensions of the coaching problem (see fig 2.3), should be used to inform the design of such a framework.

3. The coaching framework: a recommendation framework for behaviour change

This chapter presents our proposed approach to the "recommendation system for behaviour change" problem and the corresponding developed framework [115, 116]. As discussed in chapter 2 this problem is at the intersection of diverse approaches in the literature and necessitates the formalization of a new framework, integrating the constraints of personalization, repeated interaction, long-term follow-up and persuasiveness. We propose in this chapter a framework, which we call the *coaching framework*, to encompass these constraints. Diverse methods and approaches of the investigated literature on related problems inform the design of this framework. In Section 3.1, we formally describe the problem of coaching and propose a model of the task. We discuss in Section 3.2 the possible evaluations of a coaching recommendation algorithm and propose a method to measure the behaviour change at the user level. We then present an illustrative example of coaching in a simplified environment in Section 3.3 to highlight the questions raised by the coaching framework and study it analytically to draw general conclusions on critical stakes. We propose in Section 3.4 an optimal criterion for recommendations in the coaching scenario and discuss some heuristics than can be derived from it. In Section 3.5, we evaluate the proposed framework on a task simulated from real consumption data. We finally conclude on the proposed approach in Section 3.6.

3.1. Problem formalization

As stated in Chapter 1, the first question to be addressed in this thesis is how to design a recommendation framework that promotes long-term behaviour change of its users. As presented in chapter 2, we can formulate our problem as a problem of persuasive design. In 2.3.1.2, we underlined the importance of considering a valid user behaviour model when

designing a persuasive system. It then seems fundamental to inform the design of our recommender system.

In Section 2.3, we particularly focused on the *habit alteration model* defined by [104]. Indeed this model fits well the objective of long-term and lasting behaviour change. As defined by the authors, behaviour is driven by both conscious and automatic processes that result in human action. The authors argue that a way of changing behaviour is to lead the user to modify his or her habits or to develop new ones. This fits well the objective of long-term and lasting behaviour change: once habits are modified or created, they are part of the behaviour. They can even become an automatic response to a given context.

That is why we focus on habit formation and alteration in this chapter. So we focus on recommendations that target a repeated behaviour, which matches both the *habit alteration model* and the global question of this thesis work.

We then consider a scenario where a user has to face a repeated decision-making problem among a set of possible choices. It can be, for example, food items when considering the problem of meal composition, types of shots when considering the problem of playing tennis or chord sequence when facing the problem of piano improvisation.

3.1.1. The coaching scenario

We propose to model the repeated decision-making task of the user as a sequential choice of a combination of items in a set of items I. At each step t, the user chooses a combination of items, according to what we call his or her *decision function*.

Given this task, the user has the objective of changing his or her behaviour. For example, he or she wants to become better at improvising when playing the piano or to develop healthier eating habits when composing meals. To do so, he or she may call on an automated recommendation system, which role is to accompany this behaviour change. In the following, we refer to this system as a "coaching system" given that its task is to help a motivated user to reach a self-determined goal.

We propose to formalize the task of the coaching system as a recommendation task: the objective of the system is to make recommendations that will help the user to change his or her behaviour towards a targeted one. In this setting, by contrast to the classical recommender system approach, the user is not only seen as a potential consumer but as a motivated agent with an active part in the recommendation. Formally we can define the

objective of both the coaching system and the user as introducing a change in the user decision function so as to progress towards a pre-determined targeted behaviour. The role of the coaching system, then, is to make recommendations to lead the user to modify its decision function.
In the following, and for simplicity of exposure, we suppose that the user's task is to choose a single item in the item set I at each step t, instead of a combination.

3.1.2. An iterated two-player game

Given the presented scenario, how should be modelled the interaction between the coaching system and the user? We point out three key determinants:

- A specificity of the *coaching* perspective is the user's involvement in the interaction. As stated in the general definition of coaching presented in [10] the user is "perceived to be a mature, motivated, voluntary, and equal participant in a learning relationship with a facilitator whose role is to aid the learner in the achievement of his or her primarily self-determined learning objectives". Therefore, we propose to base the coach's recommendations on the user's observed behaviour. As a tennis coach who reacts to the actions of his/her pupil, either by encouraging him/her or by suggesting changes in his/her play, we consider *mistake correcting* (as defined in [71]) as the action mode of our automated coach. At each step, after the user has made a choice, the coach either agrees with this choice or makes a recommendation. This recommendation takes the form of a suggestion of substitution $i \rightarrow j$. That is, when the users choose item i, the coach recommends replacing i with j. The choice of this mode of action for the coach has several advantages. First, it meets the very concept of coaching as the recommendation is based on the user's choices: the user actively participates in the recommendation process. Second, *mistake correcting* has been proven, in the teacher-student literature, to be one of the most efficient advice distribution methods when considering agents advising agents [71], in particular for automated agents advising humans [75]. Finally, it meets the idea of *habit alteration* described in [104]. By making recommendations associated with a particular context, represented here by the user's proposed choice, the coach has greater chances to have them integrated by the user as responses to this context, and so eventually for the user to form new habits.

- Second, we emphasise the need for personalisation in the coach's recommendations. Indeed it has long been proven that personalisation [14] is a key factor in the effectiveness of recommendation systems. Since the coaching framework is built to produce recommendations, it seems fundamental to take into account the particular preferences and needs of each user in the interaction. Thus we propose to incorporate in the interaction scenario an updating phase for the coach, allowing the latter to adapt its recommendations to the particular user it faces.

- Reciprocally to the adaptation of the coach, we assume that accepting a suggestion results in learning for the user. In other words, when the user accepts to choose a given item that the coach has suggested, he or she integrates the suggestion and is more likely to choose it on his/her own in the future. The idea behind this assumption is that if he or she accepts a suggestion of substitution $i \to j$, the user is more ready to choose j instead of i in an equivalent context in the future. This assumption results from the *habit alteration* model: after being repeatedly accepted by the user in a given context, the choice of item j become a new habit of the user. This learning supposedly depends on the user's personal characteristics and the given suggestion.

Given the highlighted particularities of coaching, we propose to model the coaching interaction as an iterated two-player game. We denote the coach C as the first player and the user U as the second player.
We consider a scenario consisting of three stages, described in the following :

1. U submits a proposal to C. This proposal is the first choice of U regarding his or her decision function. Given our formalization of coaching, the proposal is an item $i \in I$.

2. C analyses U's proposal, and suggests a modification, if judged useful. We assume that this suggestion is in the form of a substitution $i \to j$.

3. U accepts or rejects the suggestion made by C. Then two outcomes are possible :

    - If U accepts the suggestion, that is replacing item i by item j, it results in a change in the decision function of U, according to his/her learning capacity, in order to propose the recommended item more frequently in the future.

- If U rejects the suggestion of C, he or she does not learn and the decision function remains unchanged.

Depending on the user feedback (i.e. acceptance or rejection), the coach can adapt its recommendations for future steps.

We consider this game iterated, as the user and the coach repeatedly follow this interaction schema during a certain number of steps $t \in T$. At each step, the interaction scenario is played, and both U and C can learn from their respective exchanges. Formally U and C are modelled as agents with their own characteristics.

3.1.3. Model of the user

We have presented in section 3.1.2 the model of interaction between the user and the coaching system. In the following section, we address the specific question of the user model. The user model is expected to be representative of the behaviour of a user engaged in a coaching interaction. Thus, it should consider the particularities of the coaching interaction as described and encompass the three main actions of the user: the proposal of an item, the acceptance or rejection of the system suggestion, and the choice function update. However, our proposed model must remain as simple and interpretable as possible. Indeed our objective is not to design a hypothetical perfect model of human behaviour but rather to test the proposed scenario in a simple case. Therefore we propose a model of the user based on three components :

- A choice vector, or probability distribution over the set of possible items :

$$\Pi_t : \begin{cases} I \to [0,1] \\ i \mapsto \pi_t(i) \end{cases}$$

It represents the preferences of U at each step and determines his or her choices. The decision function of U is represented at each step as a random draw in I following the probability distribution Π_t. As one can notice, this distribution is parameterized by t given that it can change over time.

- A matrix $\mathbf{M}_t : I \times I$ associating to each couple of item $(i,j) \in I^2$ and at each step a substitutability coefficient $m_t(i,j) \in [0,1]$. $\mathbf{M}_t$ represents the acceptability, for U,

of the possible substitutions at each step. Thus the coefficient $m_t(i,j)$ represents the probability, if item i is proposed by U at step t, of the suggested substitution $i \to j$ for being accepted by U. If the suggestion is not accepted, with probability $1 - m_t(i,j)$, U stays with his or her choice.

- A *propensity to modify* Π_t when U accepts a substitution $i \to j$ suggested by C. This accounts for the effects of the recommendation on U habits and future choices. We propose a model for this change propensity in the following form:

$$\begin{cases} \pi_{t+1}(i) = (1-\lambda)\pi_t(i) \\ \pi_{t+1}(j) = \pi_t(j) + \lambda\pi_t(i) \qquad \text{If } i \to j \text{ has been accepted} \\ \pi_{t+1}(k) = \pi_t(k)\ ,\ \forall k \in I \setminus \{i,j\} \end{cases} \tag{3.1}$$

  where $\lambda \in [0,1]$. If U does not accept the proposed substitution, he or she does not change the preference vector and $\Pi_{t+1} = \Pi_t$. This formula guarantees that if Π_t is a probability distribution, so is Π_{t+1}. In addition, one can notice that in a situation where $i = j$, i.e. a no change, this formula leads to $\Pi_{t+1} = \Pi_t$, i.e. no change in U habits. The introduced λ is a parameter that controls the strength of U behaviour change induced by C. It can be seen as the learning rate of U. The closer to 1 the value of λ, the more significant the effect of adopting a recommendation $i \to j$ on the future choices of U.

  When $\lambda = 0$ U's propensity of change is null and his/her probability distribution remains unchanged all along the interaction, regardless of the proposals made by C. In this case we have : $\forall t \in T,\ \Pi_t = \Pi_0$.

  On the other hand, when $\lambda = 1$, there is a complete transfer of the probability of choosing i to the probability of choosing j.

  In the following, we note $f_{i\to j}(\Pi)$ the preference vector of U after, starting from Π, he or she has accepted the suggestion $i \to j$ from C.

3.1.4. Model of the coach

The coach and how it interacts with the user are the core of the proposed framework. As we have stated, the coach should be able to make suggestions that are personalized and focused on long-term behaviour change.

We introduce the notion of *strategy* for the coach. The strategy c_t of a coach is a function that at each step t and for each proposal $i \in I$ of the user, associates a recommendation in the form of an item $c_t(i) \in I$ representing the substitution $i \to c_t(i)$:

$$c_t : \begin{cases} I \to I \\ i \mapsto c_t(i) \end{cases}$$

The objective of the coach being to introduce changes in the user behaviour to accompany the latter towards a targeted behaviour, its strategy c_t should reflect this objective. Finding the best strategy to do so is the subject of Section 3.4. Moreover, comparing strategies underlies the ability to measure the impact of the recommendations on user behaviour. This is the subject of the next Section 3.2.

When introducing the scenario of coaching, we underlined the importance of personalization in the recommendations, and so the importance for the coach to update its recommendations accordingly to the user feedback. Thus, we consider here an updating function g, that given the strategy c_t of the coach at step t, the user initial choice $i \in I$ and the actual choice after recommendation $j \in \{i, c_t(i)\}$, updates the strategy of the coach. We note $c_{t+1} = g(c_t, i, j)$. The problem of finding an efficient updating function g is discussed in Section 3.4.

Given the characteristics of both the user U and the coach C, the iterated two-player game is finally described in Algorithm 1.

In the rest of the chapter, we assume that the matrix $\mathbf{M}_t$ that controls the acceptability of suggestions $i \to j$ by U is constant over time, hence the notation $\mathbf{M}$.

3.2. Coaching evaluation

Once the interaction framework is defined, we are interested in evaluating coaching approaches. Indeed, as we are investigating the question of what makes an efficient coach and how to produce recommendations that efficiently promote long-term behaviour change, we need to compare coaching algorithms between them and so need to define performance metrics for a coaching interaction.

Evaluating the quality of a coaching system, i.e. a system whose goal is to promote long-term behaviour change, is a delicate question. This is due to diverse factors, the most significant of which is the difficulty of defining an appropriate horizon for behaviour quality

begin
  $t = 0$
  while *coaching in play* **do**
    $t \leftarrow t + 1$
    Decision making phase
      U chooses item i according to policy Π_t.
      C suggests substitution $i \rightarrow c_t(i)$ according to the strategy c_t.
      U accepts the substitution $i \rightarrow c_t(i)$ with probability $m_{i,c_t(i)}$.
    Learning phase for the user
      U changes the preference vector: $\Pi_{t+1} \leftarrow f\big(\Pi_t, i, c_t(i)\big)$
      (f defined according to Eq. 3.1)
    Learning phase for the coach
      C changes its strategy: $c_{t+1} \leftarrow g\big(c_t, i, j\big)$
      (see Section 3.4)
  end
end

Algorithm 1: The two-player game between U and C

measurement and the very definition of behaviour quality. Considering the literature on related problems to this question is of little help.

As stated in [29], evaluation in multi-stakeholder RS is still an open problem. In addition, as presented in section 2.1.3 value-aware RS are often considering only immediate gain instead of long-term metrics. It is noticeable that the value of items and the corresponding notion of utility is, in essence, an indication of recommendation quality. Utility is then used both for discrimination among possible recommendations and for evaluation of the whole RS algorithm.

For its part, the teacher-student learning evaluation appears complex as several metrics are necessary to encompass the diverse dimensions of performance in this framework (See 2.2.3.2). However, all classical metrics are based on the student's performance on the considered task. This is easy to determine when assuming the student is a reinforcement learning agent performing a rewarded task. Thus in the case of coaching, it points out the need for an evaluation of learner behaviour on the task to evaluate the coach properly.

So when considering approaches of the literature, the existing approaches involve a measure of the performance induced by the system at the user level (either the utility or the user reward). Thus we propose a similar approach for coaching with a user-level measure of performance.

3.2.1. The concept of score

As presented in section 3.1.1, the user's task in our proposed coaching framework is to sequentially choose items among a set I.
Inspired by possible application domains of coaching, we propose to evaluate the user-level performance through a score function distributed over the items, defined as follows:

$$S : \begin{cases} I \to \mathbb{R} \\ i \mapsto s(i) \end{cases}$$

It is noticeable that many problems can be modelled by such a score function. For example, in healthy eating, one can associate a nutritional score with food items, while in tourism recommendation systems, one can associate a carbon footprint with each possible choice. In essence, such a score function can be considered as the utility of the possible items regarding the pursued objective of coaching.
Given this formalization, and the fact that a U is characterized by the probability distribution Π_t over I, we can express the mean score value of Π_t as:

$$\mathcal{V}(\Pi_t) = \sum_{i \in I} \pi_t(i) \cdot s(i) \tag{3.2}$$

The objective of the coaching strategy in this setting is to improve this mean value as much as possible in the shortest possible time through recommendations of items by C to U. That is to make the user U follow a trajectory in the space of preferences, i.e. the space of the possible probability vectors Π_t.

3.2.2. Coaching performance metrics

The introduced notion of score makes it possible to compute a metric of the user behaviour quality: $\mathcal{V}$. The problem we face now is to measure the performance of a given coaching algorithm interacting with a given user on a task. As stated below, the metrics in the multi-stakeholder RS literature are not well adapted to coaching as they do not consider efficiently the long-term behaviour change of the user. On the other hand, metrics from teacher-student learning literature are based on a comparison of students' learning with and without a teacher. However, in our proposed framework, the user U is not informed of the score: we assume that he or she cannot judge by himself/herself the "quality" of

items regarding the pursued objective of behaviour change. Otherwise, he/she would not need the coach. Therefore he or she is not able to learn on his/her own how to maximize it, as a reinforcement learning agent would, having no notion of reward. So it does not make sense to compare a single user that does not modify his/her behaviour to a coached one. A possible approach to overcome this could be to compare the behaviour of the coached user to the optimal user behaviour given the score function S.
Consider the space of all possible probability vectors of the form Π, denoted $\mathcal{P} \subset [0,1]^I$. Given that space, we note $\Pi^\star$ the set of probability distributions in $\mathcal{P}$ that are associated with the maximal value of $\mathcal{V}$: $\Pi^\star = \text{ArgMax}_{\Pi \in \mathcal{P}} \mathcal{V}(\Pi)$.
Let us note Π_0 the starting preference vector of the user U. Given the characteristics of each peculiar user, that is the acceptability matrix $\mathbf{M}$ and the learning rate λ defined in the user model (3.1.3), he or she is potentially only able to reach a subspace of the total space of preference vectors $\mathcal{P}$ when starting from Π_0. For example, it may exist an item k in I such that $\forall\, i \in I \setminus \{k\}$: $m_t(k,i) = 0$, hence the incapacity for U to reach any vector Π_t where $\pi_t(k) > \pi_0(k)$. We note the resulting subspace $\mathcal{P}_\mathsf{U} \subseteq \mathcal{P}$. As a consequence, the set of vectors that maximize $\mathcal{V}(\Pi)$ and are reachable from Π_0 may be different from $\Pi^\star$. We note it $\Pi^\star_\mathsf{U} = \text{ArgMax}_{\Pi \in \mathcal{P}_\mathsf{U}} \mathcal{V}(\Pi)$. Given that, we propose the following metrics to evaluate the performance of a coaching strategy:

1. The first considered metric stresses the *level of performance that one wants to obtain* $\eta\mathcal{V}(\Pi^\star_\mathsf{U})$ with $\eta \in (0,1)$ and measures the mean number of interactions that the coach needs to guide the user towards this performance level: $\overline{T}_\eta$ starting from Π_0.

   This can be seen as an adaptation of the *time to threshold* metric presented in 2.2.3.2. The threshold here is defined as a portion η of the optimal user behaviour value. However, this metric suffers from some limitations. First, it is difficult to evaluate $\Pi^\star_\mathsf{U}$, the optimal user behaviour. Second, $\Pi^\star_\mathsf{U}$ depends by essence on Π_0, $\mathbf{M}$ and λ. Therefore it can be highly variable between users and may be more adapted when comparing coaching algorithms on a single user rather than on a population.

2. A second approach is inspired on the notion of *budget* defined in the teacher-student learning literature (see 2.2.3.1). Here, as U only learns when interacting with the coach, we consider the *budget* T to be the total interaction time. A dual measure of performance in this setting is the *mean gain of performance* $\overline{\mathcal{V}}_T = \text{mean}\big(\mathcal{V}(\Pi_T) - \mathcal{V}(\Pi_0)\big)$ after T interactions. The mean here is taken from repeated episodes of T interactions starting from Π_0 since an episode stems from stochastic choices from

the user.

3. Our third proposed approach is to consider a *criterion based on the whole trajectory* in the preference vector space. This is typically the case for the metric based on *total reward* proposed in the teacher-student learning literature (see 2.2.3.2). In our setting, as U is not directly rewarded, we can consider, for instance, a cumulative gain: $G(T) = \sum_{t=1}^{T} \big(\mathcal{V}(\Pi_t) - \mathcal{V}(\Pi_0)\big)$.

In the following, we will focus on the second criterion $\overline{\mathcal{V}}_T$. It allows easy comparisons, especially in the case of a real user having a limited number of interactions with the system: as stated in [71], the notion of *budget* is of critical importance when interacting with human agents having limited attention.

3.3. Analytical study of a simple case

In this section, we present a simplified coaching case. We propose an analytical study of this case to raise the main issues of the recommendation problem faced by the coach. In particular, we show how the optimal recommendation strategy of the coach depends on the user characteristics.

3.3.1. Definition of the studied scenario

We consider an item set I composed of three elements:

$$I = \{i_1, i_2, i_3\}$$

We define the associated score function S as:

$$S : \begin{cases} I \to \mathbb{R} \\ i_1 \mapsto s(i_1) = 5 \\ i_2 \mapsto s(i_2) = 20 \\ i_1 \mapsto s(i_1) = 50 \end{cases}$$

We consider for the following user U initial choice vector:

$$\Pi_0 = (1, 0, 0)^\top$$

which means that U always chooses i_1
The user acceptability matrix is denoted $\mathbf{M}$ and his or her learning rate is denoted λ:

$$\mathbf{M} = \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} \\ m_{2,1} & m_{2,2} & m_{2,3} \\ m_{3,1} & m_{3,2} & m_{3,3} \end{pmatrix} \quad \text{and} \quad \lambda \in [0, 1]$$

The environment of the studied coaching problem is depicted in figure 3.1.

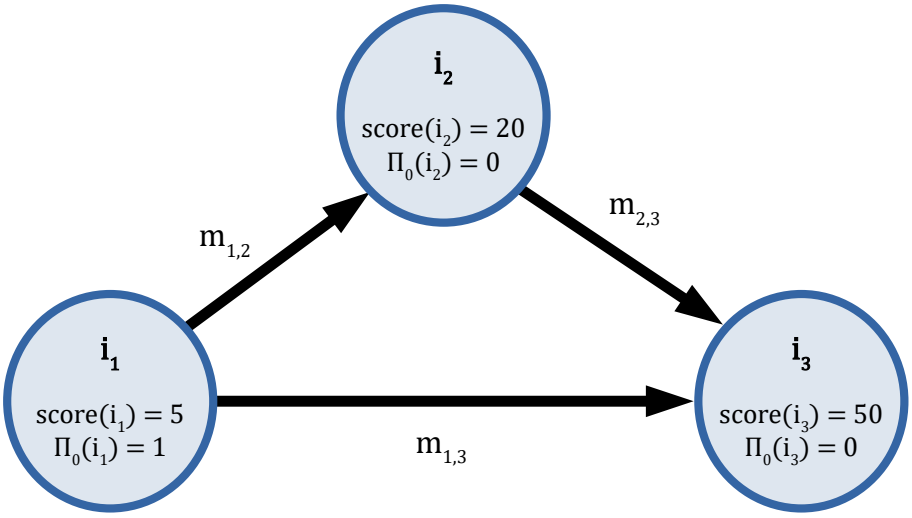

Figure 3.1: Presentation of the studied simplified case at $t = 0$.

3.3.2. The problem of item recommendation

In this case, we assume that $\forall (i, j) \in I^2 \; m_{i,j} > 0$. Accordingly, one can compute the optimal choice vector $\Pi^\star_{\mathsf{U}}$ of U and the associated expected score $\mathcal{V}(\Pi^\star_{\mathsf{U}})$. We have :

$$\Pi^\star_{\mathsf{U}} = (0, 0, 1)^\top \quad \textit{and} \quad \mathcal{V}(\Pi^\star_{\mathsf{U}}) = 50$$

The coach's problem in this setting is finding the best recommendation for each possible proposal made by U. This problem can be reformulated as the problem of finding the optimal recommendation strategy $c^\star$.

The best recommendation strategy in the presented case should lead the user from Π_0 to $\Pi^\star_{\mathsf{U}}$. For both user proposals i_2 and i_3 the solution is trivial given the score function S: $c^\star(i_2) = i_3$ and $c^\star(i_3) = i_3$. Conversely, the problem of finding the best recommendation $c^\star(i_1)$ when U proposes i_1 is not easy since two paths are possible towards the best item i_3.
Let us consider the performance associated with each possible recommendation:

1. The first possibility is $c^\star(i_1) = i_1$, i.e. for the coach not to make any recommendation. Given the considered learning model of U, this does not lead to any change in the user preference vector. Thus it does not lead the user towards $\Pi^\star_{\mathsf{U}}$.

2. The second possibility is that the coach recommends replacing i_1 by i_3, that is $c^\star(i_1) = i_3$. This strategy can be qualified as the "direct path strategy" when considering figure 3.1. The expected score of the preference vector of the user at step $t+1$ regarding t is given by the following expression:

$$\begin{aligned} \forall t \in T:\ \mathbb{E}[\pi_{t+1}(i_1)] &= (1 - m_{1,3})\,\pi_t(i_1) \ +\ m_{1,3}(1-\lambda)\,\pi_t(i_1) \\ &= \pi_t(i_1) \ -\ m_{1,3}\,\lambda\,\pi_t(i_1) \\ &= \pi_t(i_1)\,(1 - m_{1,3}\,\lambda) \end{aligned}$$

   Given that we can compute the expected score of the user preference vector at each step:

$$\forall t \in T\ :\ \mathbb{E}(\pi_t(i_1)) = \pi_0(i_1)\,(1 - m_{1,3}\lambda)^t$$

   In this case, the only items possibly considered by U are i_1 and i_3. Therefore we have:

$$\forall t \in \{0, \ldots, T\}: \begin{cases} \mathbb{E}[\pi_t(i_1)] = (1 - m_{1,3}\lambda)^t \\ \mathbb{E}[\pi_t(i_2)] = 0 \\ \mathbb{E}[\pi_t(i_3)] = 1 - (1 - m_{1,3}\lambda)^t \end{cases} \tag{3.3}$$

3. The third possibility for the coach is to let the user follow an "indirect path" to i_3, that is to recommend $i_1 \to i_2$ when the user proposes i_1 and $i_2 \to i_3$ when the user

proposes i_2.

Following the same method we have :

$$\mathbb{E}[\pi_t(i_1)] \;=\; \pi_0(i_1)\,(1 - m_{1,2}\lambda)^t$$

In addition :

$$\begin{aligned}\mathbb{E}[\pi_{t+1}(i_2)] \;&=\; \pi_t(i_2) \;+\; m_{1,2}\,\lambda\,\Pi_t(i_1) \;-\; m_{2,3}\,\lambda\,\pi_t(i_2)\\ &=\; \pi_t(i_2)\,(1 - m_{2,3}\,\lambda) \;+\; m_{1,2}\,\lambda\,\pi_t(i_1)\end{aligned}$$

which is the discrete version of the following differential equation:

$$\frac{\mathrm{d}\,\pi_t(i_2)}{\mathrm{d}t} \;=\; -\ln(1 \;-\; \lambda\,m_{1,2})\,\pi_t(i_1) \;+\; \ln(1 \;-\; \lambda\,m_{2,3})\,\pi_t(i_2)$$

For which the solution is known:

$$\pi_t(i_2) \;= \pi_0(i_2)\,(1 -\; \lambda\,m_{2,3})^t \quad + \pi_0(i_1)\,\frac{-\ln(1 - m_{1,2}\,\lambda)\big((1 - m_{1,2}\,\lambda)^t - (1 - m_{2,3}\,\lambda)^t\big)}{-\ln(1 - m_{2,3}\,\lambda) + \ln(1 - m_{1,2}\,\lambda)}$$

As in this case $\pi_0(i_2) = 0$, we finally have:

$$\forall t \in \{0, \ldots, T\} : \begin{cases}\mathbb{E}[\pi_t(i_1)] \;=\; \Pi_0(i_1)\,(1 - m_{1,2}\lambda)^t \;=\; (1 - m_{1,2}\lambda)^t\\ \mathbb{E}[\pi_t(i_2)] \;=\; \frac{-\ln(1-m_{1,2}\,\lambda)\left((1-m_{1,2}\,\lambda)^t-(1-m_{2,3}\,\lambda)^t\right)}{-\ln(1-m_{2,3}\,\lambda)+\ln(1-m_{1,2}\,\lambda)}\\ \mathbb{E}[\pi_t(i_3)] = 1 - \mathbb{E}[\pi_t(i_1)] - \mathbb{E}[\pi_t(i_2)]\end{cases} \tag{3.4}$$

As we can see, the three possibilities lead to different outcomes. The question for the coach then is to choose the best possible recommendation strategy regarding its performance metric $\overline{\mathcal{V}}_T$. Given the computed preference vectors, we can notice that the respective performance of the investigated recommendation strategies will depend on λ, T, $m_{1,2}$, $m_{2,3}$ and $m_{1,3}$. Let us consider the given instantiation of the substitutability matrix:

T	λ_c
10	1
20	≈ 0.6028
50	≈ 0.2615
100	≈ 0.1346
1000	≈ 0.0138

Table 3.1: λ_c values regarding T.

$$\mathbf{M} = \begin{pmatrix} 1 & 0.15 & 0.1 \\ 0 & 1 & 0.2 \\ 0 & 0 & 1 \end{pmatrix}$$

We can then easily compute the difference between the performance of the two presented strategies (Eq. 3.3 and Eq. 3.4). The optimal strategy, in this case, depends on λ, the user learning rate, and T, the total number of interactions. Here it exits a critical value λ_c for λ such that if $\lambda > \lambda_c$, the "indirect path" strategy is better regarding $\overline{\mathcal{V}}_T$ than the "direct path" strategy.
In this case, we can see (table 3.1) that the higher T, the lower λ_c. Therefore an efficient estimation of U parameters by C makes it possible to choose the best recommendation.

3.3.3. Conclusion

The analytical study of this simplified case raises some important questions for the design of an efficient coaching algorithm.
First, it underlines the fact that the best recommendation strategy for the coach may be non-myopic. In our case, depending on the characteristics of the user, the best recommendation strategy through the optimum cannot be inferred from the direct expected score gain. It exists cases (represented here by the "indirect path" strategy) where trying to maximize the instantaneous expected score gain leads to sub-optimal recommendations.
Second, it emphasizes the importance for the coach to personalize its recommendation to the user. Essentially, we have seen that the characteristics of a given user affect the best possible recommendation strategy for this user. In the presented case, inferring strategies was pretty straightforward, considering the small number of items and so, of possible recommendation trajectories. In addition, we assumed that the coach is perfectly informed about the user. Conversely, a coaching interaction in a realistic case involves

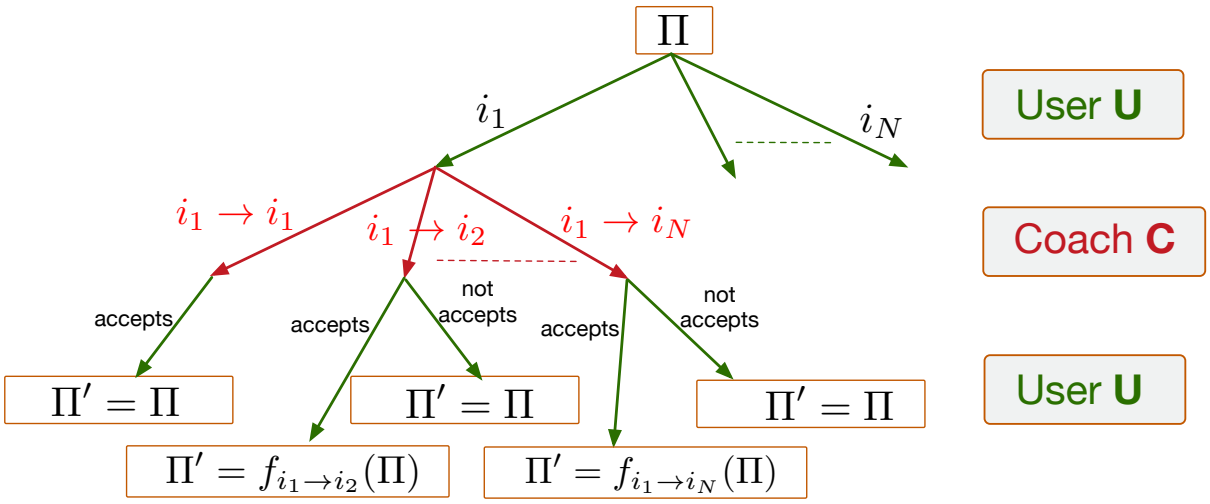

Figure 3.2: A decision step faced by the coach. Following his/her preference Π_t, the user chooses one item, and then the coach must select a substitution, which, in turn, can be accepted or rejected by the user. After this turn, the preference vector is updated.

a larger item set, thus a tremendously greater number of possible trajectories. It also underlies a coach that is not a perfectly informed oracle and needs to learn the best recommendation strategy from its interactions with the user, for example, by maintaining an estimate of U's characteristics.

The question of how to efficiently learn relevant recommendation strategies from the interaction with the user is addressed in the next section.

3.4. The space of coaching strategies

The task of the coach, as we defined it is to suggest to a user U a substitution $i \to j$ based on the user proposal i at each step, with the objective of leading U towards a better (regarding the score function) behaviour. This includes the identity substitution $i \to i$, corresponding to a case where the coach is satisfied with the user's choice and does not make any recommendations.

Conversely to the simple case presented in section 3.3, in the general setting, the coach has no access to the optimal preference vector of the user $\Pi^\star_{\mathsf{U}}$. Indeed, computing $\Pi^\star_{\mathsf{U}}$ necessitates a perfect knowledge of the user's characteristics Π_0, $\mathbb{M}$ and λ (see 3.2.2), which is not assumed in the case of a realistic coaching interaction. Thus the coach has to *learn* from its interactions with the user a recommendation strategy.

We introduce $V(\Pi)$, which represents the desirability that the user is in state Π from the perspective of the coach who looks at long-term expected benefits if the coach follows the optimal policy defined below by Equation 3.5. Then, for each possible choice of item $i \in I$ by U, the coach should choose the substitution $i \to j^\star$ such that:

$$j^\star \;=\; \underset{j \in \mathcal{I}}{\text{ArgMax}} \Big\{ m_{i,j} \left[\big(s(j) - s(i)\big) \;+\; V\big(f_{i \to j}(\Pi)\big) \right] \;+\; (1 - m_{i,j})\, V(\Pi) \Big\} \tag{3.5}$$

Now, the expected value $V(\Pi)$ of all preference vectors Π are fixed point of the Bellman equation that relates the updated evaluation $V_{t+1}(\Pi)$ with the current evaluations of the preference vectors $V_t(\Pi')$ that may ensue a recommendation by the coach (see Figure 3.2).

$$V_{t+1}(\Pi) \;=\; \sum_{i \in \mathcal{I}} \pi(i) \; \underset{j \in \mathcal{I}}{\text{Max}} \Big\{ m_{i,j} \left[\big(s(j) - s(i)\big) + V_t\big(f_{i \to j}(\Pi)\big) \right] \;+\; (1 - m_{i,j})\, V_t(\Pi) \Big\} \tag{3.6}$$

where $f_{i \to j}(\Pi)$, the preference vector resulting from the acceptation of the suggestion $i \to j$, is defined by Equation 3.1.

These equations require that the coach knows the matrix $\mathbf{M} = [m_{i,j}]$ $(1 \leq i, j \leq |I|)$ and the current preference vector Π of the user, as well as the learning rate λ to compute $f_{i \to j}(\Pi)$. Thus the optimal choice criterion presented in Equation 3.5 is not directly applicable in a scenario where we assume that the coach does not have prior knowledge of the user at hand.

But from this optimal criterion, we can derive heuristic ones and the subsequent strategies. To do so, simplifications in the optimal criterion are made, and/or factors are ignored.

Strategy Greedy Score (GS)

The simplest strategy for the coach is to ignore all the characteristics of the user and to consider only the score function. That is to suggest at each interaction the substitution $i \to j$ associated with the highest score gain: $s(j) - s(i)$.

$$j^\star \;=\; \underset{j \in \mathcal{I}}{\text{ArgMax}} \big[s(j) - s(i) \big] \tag{3.7}$$

Strategy Greedy Expected Score (GES)

A second strategy takes into account the acceptability matrix $\mathbb{M}$ of the user but ignores the possible changes in the preference vector $\big($i.e. $f_{i \to j}(\Pi_t) = \Pi_t\big)$, which gives:

$$j^\star \;=\; \underset{j\in\mathcal{I}}{\text{ArgMax}}\Big\{\, m_{i,j}\,\big[s(j)-s(i)\big]\,\Big\} \tag{3.8}$$

We call the corresponding strategy *greedy-expected-score* (GES) because it does not consider rewards beyond the immediate one.

Strategy Greedy Acceptation (GA)

This strategy maximizes the probability of the user accepting $i \rightarrow j$ as long as the corresponding change in score is positive or null: $s(j) - s(i) \geq 0$. In this way, it is hoped that the user changes his/her behaviour more easily and that, in the longer term, this will overcome a lack of high gain in short term.

$$j^\star \;=\; \underset{j\in\mathcal{I}}{\text{ArgMax}}\Big\{\, m_{i,j} \;\mid \big(s(j)-s(i)\big) \;\geq\; 0\,\Big\} \tag{3.9}$$

Both the GA and the GES strategies maintain an estimate $\widehat{\mathbf{M}}$ of the matrix $\mathbf{M}$ based on the interactions with the user. More specifically, each element $m_{i,j}$ of the matrix is evaluated using the following equation:

$$\widehat{m}_{i,j}^{t+1} = \begin{cases} \frac{\widehat{m}_{i,j}^{t}+1}{n_{i,j}} & \text{if the substitution } i \rightarrow j \text{ is accepted} \\ \frac{\widehat{m}_{i,j}^{t}}{n_{i,j}} & \text{otherwise} \end{cases}$$

with $\widehat{m}_{i,j}^{t}$ the current estimate of $m_{i,j}$ and $n_{i,j}$ the number of times the recommendation $i \rightarrow j$ has been proposed to the user.

All of *the above strategies are myopic*, in that they do not explicitly take into account the gains that a substitution $i \rightarrow j$ can bring in the long term. They do not try to estimate the values $V_t(\Pi)$, a feat that indeed requires the exploration of the possible consequences of the choice j to learn $V_t(\Pi)$ $(\forall t)$.

We thus introduce a *reinforcement learning* (as defined in [59]) algorithm in order to assess the merit of estimating longer-term gains when choosing a suggestion of substitution. This type of approach has been popularised in recommender systems to tackle the sequential nature of recommendation [117, 118].

Q-learning

The equation that evaluates the merit of suggesting j when the user has chosen i and accepts the proposed substitution is:

$$Q(i,j) \;\leftarrow\; (1-\alpha)\,Q(i,j) \;+\; \alpha\,\Big\{\big(s(j)-s(i)\big) \;+\; \gamma \operatorname*{Max}_{k\in\mathcal{I}} Q(j,k)\Big\} \tag{3.10}$$

where α controls the learning rate, and γ is a discount factor used to value short-term gains more than longer-term ones.
If the user refuses the substitution $(i \to j)$, the equation is reduced to:

$$Q(i,j) \;\leftarrow\; (1-\alpha)\,Q(i,j) \tag{3.11}$$

The Q values gradually reflect the long-term potential of the choices of substitutions.
When the user selects the item i, the coach suggests the item $j^\star$ according to:

$$j^\star \;=\; \operatorname*{ArgMax}_{j\in\mathcal{I}}\Big\{\, Q(i,j)\Big\} \tag{3.12}$$

It is important to note that this strategy does not directly use knowledge of the acceptability matrix $\mathbf{M}$. On the one hand, this avoids the necessity to estimate it and is, therefore, a more general approach to the coaching problem. On the other hand, this usually has to be paid for by a longer learning phase.

Item-Based Collaborative Filtering strategy (IBCF)

A baseline strategy is the one used in a standard recommendation scenario: the item-based collaborative filtering strategy [119]. In this approach, a similarity $sim(\cdot,\cdot)$ between the items is precomputed using the expressed choices of the users (e.g. food item consumption). Then, when the user selects an item i, the recommending system suggests the item $j^\star$ according to equation:

$$j^\star \;=\; \operatorname*{ArgMax}_{j\in\mathcal{I}}\left\{\, \frac{\sum_{n\in\mathcal{I}}\big(sim(j,n) * R_{u,n}\big)}{\sum_{n\in\mathcal{I}} sim(j,n)} \right\} \tag{3.13}$$

where $R_{u,n}$ is the rating of item n by user U (here, this rating is estimated by the consumption frequency of n by U) and the similarity is computed as usual in recommender systems ([119]).

Item-Based Collaborative Filtering strategy with score (IBCFs)

A natural question is whether a classical recommendation strategy, such as IBCF, could be tweaked in order to make recommendations aimed at changing the behaviour of the users. One simple way to do so is to modify Equation 3.13 to include the score gain associated with a recommendation:

$$j^{\star} \;=\; \underset{j \in \mathcal{I}}{\text{ArgMax}} \left\{ \frac{\sum_{n \in \mathcal{I}} \big(sim(j,n) * R_{u,n} \big)}{\sum_{n \in \mathcal{I}} sim(j,n)} * \big(s(j) - s(i) \big) \right\} \tag{3.14}$$

In this way, IBCFs will tend to recommend to user U substitutions j in $i \rightarrow j$ that are closed to the ones already consumed by U and that bring as much gain as possible.

In the next section, we propose to evaluate the presented strategies on the healthy food recommendation problem.

3.5. Experimental evaluation

3.5.1. The Experimental Protocol

We propose to evaluate the coaching framework on the healthy food recommendation task. In this setting, the coach's objective is to accompany a user toward developing healthier eating habits.

The experiments simulate interactions between the coach, using a given strategy, and users characterised by their matrix $\mathbf{M}$, a propensity to learn λ and a starting preference vector Π_0 over the available items. In order to have simulated users with realistic characteristics, we derived the latter from real data in the nutrition field as explained below.

We considered different user profiles, different strategies for the coach, and different initialisation settings. In our experiments, the number of interactions was set to $N = 2000$ to measure each coaching strategy's long-term trends. The results show that most effects are already obtained after 500 interactions or less, which appears to be realistic for a coaching scenario. All results are obtained from 200 simulations for each situation.

The coaching task we focus on in these experiments is based on the choice of one dessert by users. Indeed, as it will be discussed in Chapter 4, the food recommendation task is difficult because of the traditional organisation of food consumption in meals composed of several items. By focusing on the dessert recommendation, we propose a task that satisfies the simplifying assumption that the user only chooses one item at each step and is yet applicable in a real-world scenario.

3.5.2. The Nutritional Score

In this work, we assumed that a score could be assigned to each food item. For this, we used the nutritional score designed by Rayner and colleagues [120]. This score is calculated food item by food item, based on its composition, regarding a list of key nutrients. Thus this score completely ignores the context of consumption and the consumption history of a particular user. This type of score has been developed to compare the nutritional quality of food items that fulfil a similar role in a diet. Therefore it fits our approach to food recommendation, focused on desserts. However, as we will discuss in 4.5 other nutritional scores may be better adapted when considering a broader coaching task, such as promoting healthy eating at the diet level as a whole. In our case, we used a mapping from each food item present in the INCA2 database to the nutrients registered in the Ciqual food composition database[1], to compute a score for each food item.

3.5.3. The Simulated Users

3.5.3.1 The INCA II database

The Individual and National Food Consumption Survey (INCA2) database provides a snapshot of the food consumption habits of the population of metropolitan France gathered between 2006 and 2007[2]. It contains data about the meals consumed over a week by 4079 individuals.
From this population, we retained only the adults since children are not the primary target for food coaching. Thus we considered only users that are 20 or older. This resulted in a database containing the consumption of 2552 users and 365,621 registered meals.
As discussed in 3.5.1, we further focus on the choice of desserts among 267 possibilities. In order to get a rather homogeneous set of users, we selected women (who represent more than 80% of the respondents in the survey) over 20 years of age, yielding 1497 users.

3.5.3.2 Computing the initial preference vectors

After having selected the group of users we are interested in, we studied their consumption habits. To do so, we extracted the reported consumption for each user over a week. Then

[1] see https://ciqual.anses.fr
[2] See https://www.anses.fr/en/content/anses-food-consumption-data-made-available-open-data

we used the Rayner nutritional score, presented in 3.5.2, to compute an average nutritional value for each user. Thus, we could split the 1497 users into two sub-groups: women with "bad" nutritional habits (i.e. with an average score in the lower third among women), and one with "good" habits (the top third).
For each of these groups, we considered the observed consumption frequencies as an estimation of the real habits of the users. We then extracted the frequencies and used the resulting vectors as the initial preference vectors Π_0 of the simulated users.

3.5.3.3 Computing the matrix M of substitutability acceptance rates

For each of the sub-groups considered, a matrix $\mathbf{M}$ was estimated, representing the extent to which the corresponding users were ready to accept to substitute one item with another. We estimate $\mathbf{M}$ directly from the database of food consumptions by a set of users following the proposition of [121]. Their hypothesis is that two items are highly substitutable if they are consumed in similar contexts but not together (e.g., butter can be substituted for margarine since they are consumed in similar contexts but usually not consumed together).
Let us thus denote for an item i the context set C_i as the set of contexts in which i is a substitutable item. If $|C_i|$ is high, then i is substitutable in many contexts.
For two items i and j, the intersection of C_i and C_j: $|C_i \cap C_j|$ provides an estimate of the number of contexts in which either i or j can be found. If $|C_i \cap C_j|$ is high, then i and j are consumed in similar contexts. Denoting by $A_{i:j}$ the set of contexts of i where j appears:

$$A_{i:j} = \{c \in C_i | j \in c\} \tag{3.15}$$

The cardinality of $A_{i:j}$ denotes how j is associated with i.
Taking into account these considerations, the authors of [121] propose the following score inspired by the Jaccard index:

$$m_{i,j} = \frac{|C_i \cap C_j|}{|C_i \cup C_j| + |A_{i:j}| + |A_{j:i}|} \tag{3.16}$$

The score equals 1 when i and j appear in exactly the same contexts and de facto $A_{i:j} = A_{j:i} = \emptyset$. If i and j are never consumed in the same context, the score equals 0. The higher $|A_{i:j}| + |A_{j:i}|$ is, the higher the association of i and j and the lesser the score $m_{i,j}$.
Even though the INCA2 database represents a large survey, still uncommon in food consumption studies, it is nonetheless limited in scope. As a result, the matrix $\mathbf{M}$ computed

from it using Equation 3.16 is sparse and does not fully represent the true propensity of users to accept suggestions of substitutions. In order to remedy this, we took into account not only the score between items but also the score computed from the higher level categorization of food items in INCA2 (e.g., *chocolate brownie* belongs to the *cakes* category and the *pastries and cakes* super-category). We added the score computed for the items to the score computed for their category and super-category to obtain the substitutability from one item to another.

3.5.3.4 Learning rates of the simulated users

As the INCA2 database reports consumption from the user on their own, and without the intervention of any recommendation, we could not infer the learning rate λ of the user from the data.
Therefore we propose considering three different values for λ, representing three possible levels of user compliance. The first considered value is $\lambda = 0.2$. We found this value to be reasonably representative of the change of habits under the suggestions of a coach. In order to investigate the effect of λ on the results of the coaching interaction, we also simulated interactions with $\lambda = 0.5$ and $\lambda = 0.9$.

3.5.4. Tested strategies

We propose here to test the strategies presented in Section 3.4. Given the coaching scenario and the proposed recommendation strategies, several questions arise.

1. *What can achieve a coach which does not take into account the characteristics of the user?* Do these types of strategies fare significantly worse than strategies that adapt to the users? One can distinguish here strategies like IBCF and IBCFs that do take into account the past consumption of the user but nothing else about U, and strategies like GA and GES and their variants that maintain an estimate of the acceptability matrix $\mathbf{M}$ of U.

2. Considering *strategies that explicitly take into account (an estimate of) the matrix* $\mathbf{M}$ and possibly the learning rate λ of the user, what are the best ones? And what is the sensitivity of the performance attained in regard to the quality of the initial estimation of these characteristics? Here, we carried out experiments with simulated users with various profiles and with different initial estimates of $\mathbf{M}$.

3. Finally, *how myopic strategies*, like GA and GES, that try to maximize only the immediate gain, *fare against non-myopic strategies* like Q-learning, but which do not explicitly maintain an estimate of the characteristics of the users? We may expect that the second will prevail but at the price of lots of training. Do the experiments confirm this?

To answer these questions, we consider, in addition to the strategies proposed in 3.4, informed versions of GA, GES and trained versions of QL. Indeed, if we are interested in the efficiency of the proposed methods to learn efficient strategies from user interactions, we also want to test the relative efficiency of these strategies once the coach as completed its learning.

In this context, we considered fully informed versions of the GA and GES strategies: **iGA** and **iGES** who have a perfect knowledge of $\mathbf{M}$ and do not have to learn it.

We also considered two additional strategies: tQL-5000 and tQL-10000, which have been respectively pre-trained for $N' = 5000$ and $N' = 10000$ episodes with a prototypical user as if they had benefited from past interactions with many more users, which would likely be the case for a realistic coach. They are then used as coaching strategies in our experiments

The main characteristics of the investigated coaching strategies are summed up in Table 3.2.

In the following, we especially look at (i) the *mean gain in performance* with respect to the number of interactions, (ii) the *recommendation rate* of the coach: it should be decreasing after a while when the user does not have anything more to learn from the coach, or he/she does not accept his/her suggestions, and (iii) the *acceptance rate* of the suggestions by the user: it should increase as the coach learns the user's characteristics.

3.5.5. The Results and their Analysis

3.5.5.1 Overall results

(see Table 3.3 and Figure 3.3)

Table 3.3 provides a comparison of the benefits for the user of using the various coaching strategies when interacting 2000 times. The mean value of $\mathcal{V}(\Pi_t)$ for $0 \leq t \leq T$ and the standard deviation for each situation are reported for 200 simulations. Several conclusions appear. *First*, the potential gains for the users are a function of the quality of the starting

	Knowledge of nutritional score	Takes $\mathbf{U}$ into account	Explicit estimation of $\mathbf{M}$	Informed	Myopic
IBCF	-	√	-	-	√
IBCFs	√	√	-	-	√
GS	√	-	-	-	√
GA	inc.	√	√	-	√
iGA	-	√	√	√	√
GES	√	√	√	-	√
iGES	√	√	√	√	√
Q-Learning	√	indirectly	-	-	-
tQ-Learning	√	indirectly	-	pre-trained	-

Table 3.2: *Main characteristics of the coach's strategies.*

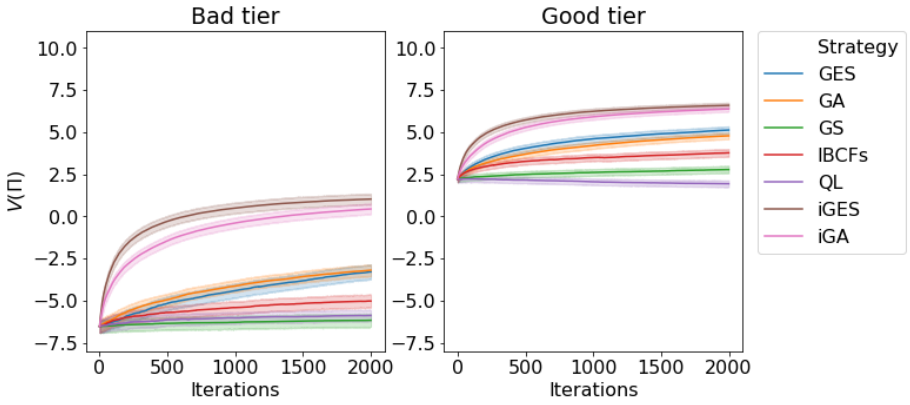

Figure 3.3: Comparison of $\mathcal{V}(\Pi_t)$, $(0 \leq t \leq T = 2000)$ for two informed strategies (iGA and iGES) and five uninformed strategies (GA, GES, IBCFs, GS and QL) for both *Bad tier* (left) and *Good tier* (right) prototype users. The colored area around the curves represent the 95% confidence interval.

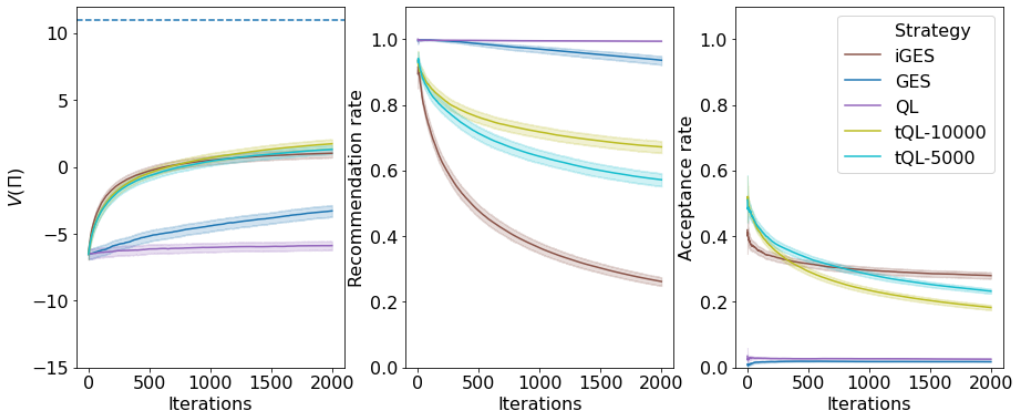

Figure 3.4: Comparison over 2000 interactions of $\mathcal{V}(\Pi_t)$, $(0 \leq t \leq T = 2000)$ (left), the recommendation rate (center) and the acceptance rate (right) for GES, Q-Learning, iGES and trained Q-Learning on 5,000 and 10,000 steps, for *Bad tier* users. The colored area around the curves represent the 95% confidence interval.

habit. As can be expected, the higher the initial quality (e.g. *Good tier* of the consumers), the lower the potential gain (see also Figure 3.3). *Second*, non-guided strategies, like IBCF, which does not take into account the nutritional score, cannot guide the user towards better habits. Even IBCFs, which does take into account the nutritional score, is inefficient because it does not consider the acceptability of substitutions by the user. GS, which only looks at the potential score's gain, is also very inefficient. Conversely, GA, which only looks at acceptability and suggests only positive substitutions, but does not consider the value of these substitutions, is surprisingly good and even better than GES on the *Bad tier* consumers. One reason may be that it tends to favour any positive move of the user, and this may accelerate changes of behaviour in the right direction as compared with GES, which tends to select the best suggestions, perhaps at the cost of their acceptability. On *Good tier* consumers, the starting preference vector of the users is better, and GES overcomes GA. *Finally*, Q-learning is good if it has benefited from previous training (see tQL-10000) and poor otherwise, which is not surprising given that Q-Learning starts with no explicit knowledge about the user. Most remarkably, tQL-10000 outperforms even iGES, which starts with perfect knowledge of the matrix $\mathbf{M}$ of the user. This is due to the non-myopic character of Q-Learning.

3.5.5.2 The behavior of the strategies

(see Figures 3.3 and 3.4)

Consumer prototype		GES	GA	IBCFs	IBCF	GS	QL	iGES	tQL-10000
Good tier	μ	2.91	2.57	1.56	1.67	0.56	-0.27	4.38	**4.49**
	σ	1.57	1.42	1.63	1.64	1.50	1.40	1.14	1.04
Bad tier	μ	3.24	3.32	1.50	1.53	0.36	0.64	7.55	**8.25**
	σ	3.11	2.48	2.80	2.86	2.84	2.41	2.29	2.46

Table 3.3: Table of the mean μ and standard deviation σ of the expected score (Eq. 3.2): $\mathcal{V}(\Pi_{T=2000}) - \mathcal{V}(\Pi_0)$, for *Good tier* and *Bad tier* consumers depending on the coaching strategy.

(1) Regarding the *recommendation rate* (see Figure 3.4), one can note that the worst strategies: QL and GES, which both have to explore possible recommendations in order to learn from the user, keep a high recommendation rate, whereas the better strategies iGES, tQL-5000 and tQL-10000 tend to quickly not to have to make recommendations since the user is rapidly improving his/her behaviour.

(2) Regarding the *acceptance rate* by the users (see Figure 3.4), it is interesting to see that iGES, which is fully informed about the matrix $\mathbf{M}$, and tQL-5000 and tQL-10000, which have been trained, have the highest acceptance rate by far. And they tend to keep it that way during the 2000 iterations, while the poorly informed strategies GES and QL make recommendations that are rarely followed by the user. It may appear that iGES, with its highest acceptance rate than tQL-2000 and tQL-10000, is better. But this is an illusion. Indeed iGES makes less recommendations, and the fewer remaining recommendations are well accepted by the users since iGES knows $\mathbf{M}$ perfectly. Conversely, the strategies tQL-5000 and tQL-10000 evolve over time, and they explore a larger space of choices by the users. Hence the recommendation rate stays high, but because these strategies do not have a perfect knowledge of $\mathbf{M}$, the acceptance rate of the more adventurous recommendations falls down more rapidly than iGES.

3.5.5.3 Influence of having prior knowledge of the user

(see Table 3.3 and Figures 3.3 and 3.4)
One important question is whether prior knowledge by the coach about the user brings a significant gain in the user's performance. Experimental results show that *it is very beneficial to have a good prior knowledge* of the user's characteristics. While it can be expected that the adaptive strategies GA and GES tend to the performances of iGA and iGES for a large number of interactions, for less than 2000 interactions, the difference in performance is striking. The same effect can be seen for Q-learning algorithms. The pre-

Consumer prototype		GES	GA	IBCFs	IBCF	GS	QL	iGES	tQL-10000
Good tier	μ	3.50	3.40	1.97	2.08	0.99	-0.67	4.67	**5.22**
	σ	1.56	1.39	1.77	1.86	1.71	1.44	1.12	0.84
Bad tier	μ	4.38	4.80	2.24	2.50	0.57	0.56	8.02	**9.57**
	σ	3.37	2.59	3.29	3.06	2.97	2.54	2.33	2.58

Table 3.4: Table of the mean μ and standard deviation σ of the expected score : $\mathcal{V}(\Pi_{T=2000}) - \mathcal{V}(\Pi_0)$, for *Good tier* and *Bad tier* consumers with learning rate $\lambda = 0.5$ depending on the coaching strategy.

trained tQ-Learning algorithms show increasing levels of performance when the number of interactions in pre-training goes from 5,000 steps to 10,000 steps.
It must be noticed that in a realistic setting, a digital coach will benefit from interactions with thousands of users simultaneously, which will provide knowledge about prototypical users and will thus result in high-quality prior knowledge. It can thus be expected that the performances obtained will tend to the higher end of the spectrum of possible performances.

3.5.5.4 Myopic vs. non myopic strategies

(see Figure 3.4)
It is expected that non-myopic strategies, like Q-Learning, outperform myopic strategies. The question is by how much. Our results confirm the advantage of these strategies. In the case of Q-learning, pre-training permits significantly overcoming the performance obtained with iGES, which has perfect knowledge about the matrix $\mathbf{M}$ of the user. This is a remarkable feat, given the high level of performance exhibited by iGES. In the context of a digital coach, these results advocate the use of reinforcement learning algorithms.

3.5.5.5 Dependence of the results over λ

Table 3.4 and figure 3.5 present results for $\lambda = 0.5$, while table 3.5 figure 3.6 present results for $\lambda = 0.9$. Table 3.4 and Table 3.5 report the mean value of $\mathcal{V}(\Pi_T) - \mathcal{V}(\Pi_0)$ and the standard deviation for 200 simulations and $T = 2000$, respectively for $\lambda = 0.5$ and $\lambda = 0.9$. The results obtained, compared with results for $\lambda = 0.2$ (Table 3.3), lead to some remarks.
First, it is noticeable that for every strategy other than uninformed Q-learning, a higher user learning rate leads to better performance. This is not surprising, considering that

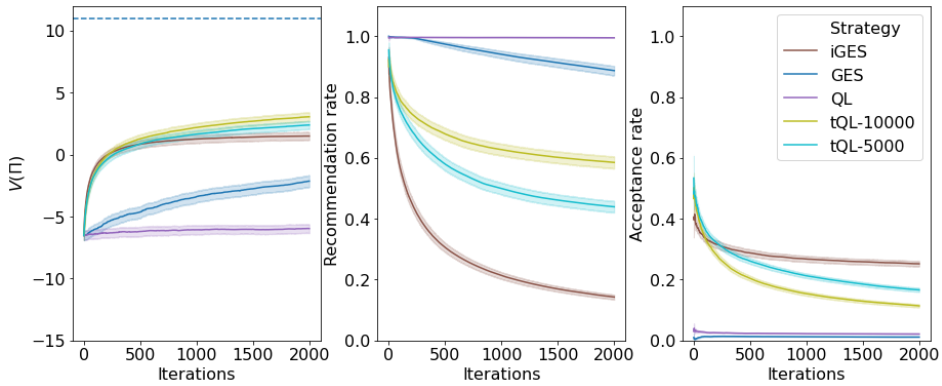

Figure 3.5: Comparison over 2000 interactions of $\mathcal{V}(\Pi_t)$, $(0 \leq t \leq T = 2000)$, the recommendation rate and the acceptance rate for GES, Q-Learning, iGES and trained Q-Learning on 5,000 and 10,000 steps, for *Bad tier* users with learning rate $\lambda = 0.5$. The colored area around the curves represent the 95% confidence interval.

Consumer prototype		GES	GA	IBCFs	IBCF	GS	QL	iGES	tQL-10000
Good tier	μ	4.20	4.39	2.05	2.40	1.41	-1.40	4.82	**5.83**
	σ	1.70	1.38	2.56	2.63	2.12	2.20	1.12	0.80
Bad tier	μ	5.94	6.67	2.50	2.83	0.86	0.82	8.19	**10.66**
	σ	4.19	3.20	3.62	3.97	3.29	2.86	2.33	2.79

Table 3.5: Table of the mean μ and standard deviation σ of the expected score : $\mathcal{V}(\Pi_{T=2000}) - \mathcal{V}(\Pi_0)$, for *Good tier* and *Bad tier* consumers with learning rate $\lambda = 0.9$ depending on the coaching strategy.

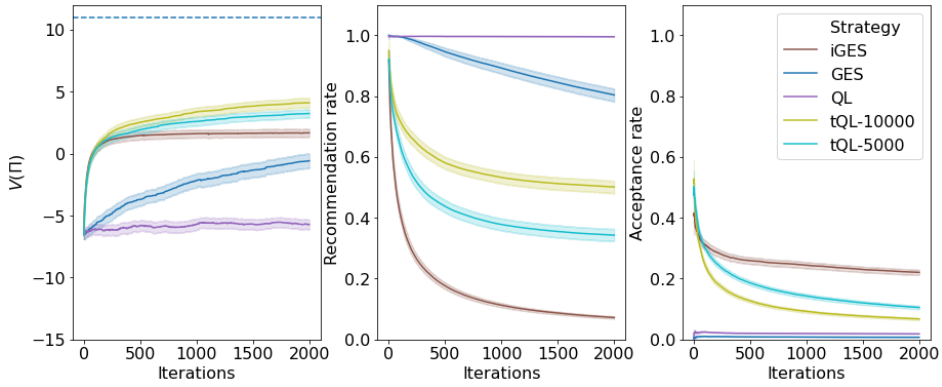

Figure 3.6: Comparison over 2000 interactions of $\mathcal{V}(\Pi_t)$, $(0 \leq t \leq T = 2000)$, the recommendation rate and the acceptance rate for GES, Q-Learning, iGES and trained Q-Learning on 5,000 and 10,000 steps, for *Bad tier* users with learning rate $\lambda = 0.9$. The colored area around the curves represent the 95% confidence interval.

a higher λ leads to faster learning from the user, allowing him/her to perform more important behaviour changes in the same time period T. While most strategies are able to quickly adapt and make useful recommendations, Q-learning, which needs much more exploration, leads the user to even worst behaviour when facing good-tier users.

Second, except for GA, which appears to be slightly better than GES on good tier users with $\lambda = 0.9$, the relative performance of strategies is mainly maintained between set-ups. Therefore, we can conclude that the strategies' performance is robust to λ. This is a noticeable result, as we have shown in Section 3.3 that the value of λ can affect the relative performance of recommendation strategies. In the investigated case of dessert recommendation, however, it appears that the value of λ has little effect on the relative performance of the tested strategies.

Third, regarding the standard deviation, one can note that the higher λ, the greater the uncertainty. It is also noticeable that this effect is observable only for uninformed strategies (GES, GA, IBCFs, IBCF, GS and QL) but not for pre-informed strategies (iGES and tQL-10000). This can be explained by the fact that, for users with high values of λ, an accepted recommendation will have much more impact on $\mathcal{V}(\Pi_t)$. In fact, as they modify more rapidly their habits, an accepted recommendation will lead to greater changes in Π_t, and mechanically on $\mathcal{V}(\Pi_t)$ given that the latter is an expected gain calculated from Π_t. While this will lead to low consequences on pre-informed strategies that propose accurate recommendations, the effect is much more significant on uninformed strategies because, in these cases, the coach is still learning and exploring and so may propose a wider diversity of items with different score values.

Regarding $\mathcal{V}(\Pi_t)$ on figure 3.4, figure 3.5 and figure 3.6, one can note that the final performance of iGES is nearly the same for the three investigated values of λ. However, an higher learning rate leads to faster convergence. On the other hand, results for both tQ-learning-5000 and tQ-learning-10000 depend on λ and show that the higher λ, the higher the final performance, which indicates that the maximum performance of tQ-learning is not reached yet. This is confirmed by the curves of the recommendation rate, which stays high even for $\lambda = 0.9$. These results confirm the advantage of non-myopic strategies in finding alternative trajectories in the space of probability vectors Π_t, preventing the user from getting stuck in a non-improvable behaviour.

3.6. Conclusion

In this chapter, we presented our approach to the problem of recommendation for behaviour change. We proposed a framework we call the *coaching scenario*, where we model the recommendation problem as an iterated two-player game. In this context, the coach agent proposes a recommendation in reaction to the user's proposals, and by suggesting substitutions to the user's choices, allows the latter to modify his habits and possibly form new ones.

We introduced the notion of *trajectory* in the space of user habits and showed the specificity of the related recommendation problem. In particular, we demonstrated in a simple case the importance of personalisation in recommendations and the necessity to consider non-myopic strategies.

From the formal study of the coaching framework, we proposed an optimal recommendation criterion and derived several heuristic recommendation strategies from it. We proposed to test these strategies to coach users simulated from real-world consumption data on the problem of making healthier dietary choices. These experiments have shown the interest of non-myopic coaching strategies in this application case, as well as the importance of having prior knowledge of the user for the coach to be efficient.

These results prove that the coaching framework is efficient in elaborating recommendations for behaviour change. Moreover, they show that the framework could theoretically be applied to the problem of healthy eating promotion. However, as we focused on simulated users, the question of the real-world feasibility of such recommendations is still to be explored. We address this question in the next chapter.

4. Coaching in the healthy food recommendation context

The problem of healthy eating, and in particular the need for a global shift towards healthier diets, has been highlighted as a major issue of our time [122]. Indeed, according to the authors, unhealthy diets "are the largest global burden of disease, and pose a greater risk to morbidity and mortality than unsafe sex, alcohol, drug and tobacco use combined". This situation calls for the development of effective and reliable solutions to inducing changes in dietary behaviour. On the one hand, such changes may be difficult, as eating habits are known to be firmly rooted [123]. On the other hand, over the last decades, another behaviour has gradually become rooted in our daily lives: the increasingly frequent and ubiquitous use of technology. In particular, the widespread use of the internet and smartphones has generalised the interactions with so-called information technologies. Given this, a possible room to tackle the stake of shifting towards healthier diets could be to use these information technologies in a perspective of behaviour change.
In the following chapter, we investigate how recommender systems, and more particularly coaching systems, could be used to induce changes in eating behaviours. In 4.1 we present related works on healthy food recommendation and their approach to the problem. In 4.2 we propose to consider the problem of healthy food recommendation as a coaching problem. In 4.3, we present the problem of cold-start and its specificity in the case of the coaching approach. We then propose a method to tackle this problem and evaluate it on real users. In 4.4, we discuss the design of a coaching system in the real-world and investigate its effect on the acceptability of recommendations. In 4.5, we look at the proposed notion of score in the coaching framework against the existing nutritional scores in the food science literature. Finally, Section 4.6 concludes.

4.1. Healthy food recommendation: an overview

With the development of information technologies and the widespread use of smartphones, many possibilities appeared for people to monitor their food consumption and find infor-

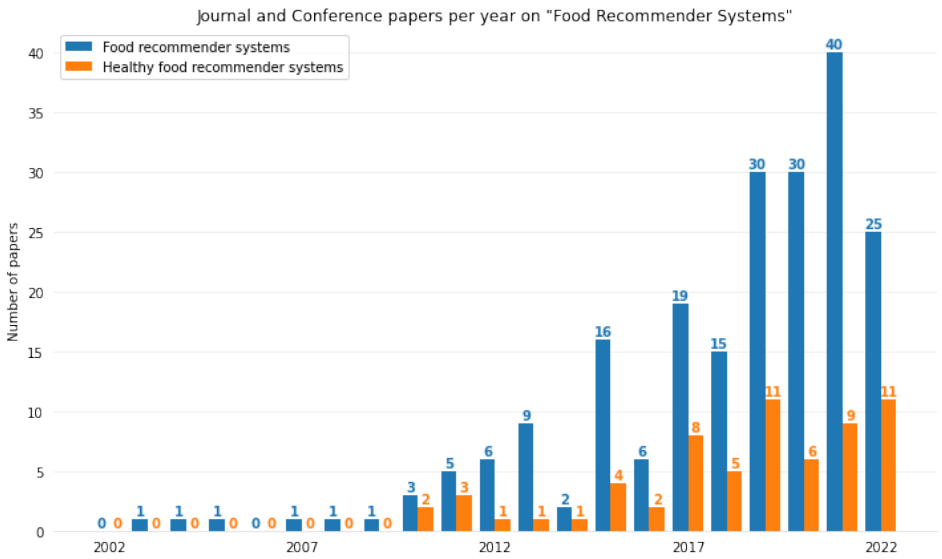

Figure 4.1: Number of both journal papers and conference papers published on the two last decades on food recommender systems and healthy food recommender systems, in the computer science field. Data comes from the Scopus database [129].

mation on food composition or new recipes. In particular, the growth of online recipe databases such as Food.com [124], Allrecipes.com [125], or Epicurious.com [126] has made readily available a wealth of food and recipe knowledge from diverse cultures around the world. This huge amount of information raises a well-known issue in information retrieval literature, the so-called *information overload* problem: it may be difficult for one to find an adapted recipe or food item corresponding to one's preferences and needs. A classical solution to address this problem is the use of Recommender Systems. Thus, the development of food recommender systems largely gained in interest over the last decade (see Figure 4.1).

But as we evoked, there are major stakes to be addressed when considering the eating behaviour of individuals, given its importance on health and disease prevention. Nevertheless, these issues are not ignored by individual consumers. Indeed, studies have shown the importance of healthiness (and particularly perceived healthiness) as a determinant of food choice [127, 128]. This opens the door for potential personal interventions promoting healthier eating habits.

Food recommender systems appear as an interesting approach when designing such intervention systems. Indeed, their intrinsic use of personalisation makes them particularly appropriate to assist people when navigating across the vast amount of available data,

helping them make satisfying and healthier food choices. In addition, their known capacity to let their users discover new interests [130, 131] made them potentially suitable in addressing the problem of inducing a shift in the user diets towards healthier ones. Thus, in recent years, research on how to encompass both user preferences and health considerations has emerged as a sub-field of the food recommender systems field. As presented in Figure 4.1, a growing body of literature is interested in the design of healthy food recommender systems. In this section, we provide a general overview of existing approaches and their results.

4.1.1. Algorithms

The first way to categorise the research work on healthy food recommender systems (RS) is to compare how the existing approaches compute the recommendations and what they take into account. That is what algorithm is used to compute the recommendation. The work in this area is very diverse, so one can find a large set of possible algorithms. Here we present only the most prominent approaches.

4.1.1.1 Collaborative Filtering

One of the main algorithmic approaches for healthy food recommendation is collaborative filtering. As presented in Section 2.1, collaborative filtering profits from data on the interest of numerous users to infer the possible suitability of a given recommendation to a given user. Collaborative filtering has been used for more than a decade in the healthy food recommendation problem [132].

For example, in order to account for the specificity of the food domain, the authors of [133] proposed a method of collaborative filtering where the similarity between users is computed by considering their health status rather than being based only on their preferences. By making groups of similar body mass index (BMI), age, weight and other user features, the problem of ratings-matrix sparsity is addressed, and the authors show there were able to make personalized healthy menu recommendations. However, the recommendations were only personalized through the prism of healthiness and did not consider the users' personal tastes.

4.1.1.2 Content based recommendation

Similarly to collaborative filtering, content-based methods are one of the most classical approaches of the RS literature. They naturally apply to recipe recommendations. Indeed, one approach frequently encountered in the literature is to consider the food items composing a recipe as its features. Given the organization of data in online recipe databases, this approach is pretty straightforward. For example, [134] proposes a content-based approach to estimate the interest of users from rated recipes. However, given the nature of food choice, expressing interest in terms of ratings may be less natural than for other goods. Moreover, the repeated nature of food choices provides extra information. Following this idea, authors of [135] propose to evaluate the interest of users in food items in terms of consumption frequencies. Another approach to content-based recommendation is referred to in the literature as *case-based recommendation*. Under this paradigm, the evaluation of a given choice is made regarding additional data, such as the context of consumption or personal food restrictions of the user. This approach has proven to be efficient in specific cases, such as recommending food for diabetic users [136]. This is particularly adapted to food recommendation, as the determinant of food choice are numerous and encompass diverse notions of context.

4.1.1.3 Hybrid recommendation

Finally, the most common approaches when considering healthy food recommendations are hybrid ones, combining collaborative filtering and content-based methods. Indeed, content-based recommendations are particularly adapted to food recommendation, given the form of the food data available and the determinants of food choice. However, it suffers from the sparsity of the observations, given the tremendous diversity of food items and, therefore, the exponential number of recipes (i.e. combinations). Given that, collaborative filtering uses the power of the community to infer the interest of users on unknown items and thus can help to reduce the sparsity of the data. Hybrid recommendations can also profit from other approaches we will not focus on here, such as demographic or location filtering. For an in-depth review of recommendation algorithms for food RS, refer to [137].

4.1.2. User modelling

The problem of data acquisition is core in the food RS domain. Indeed, in contrast to other recommendation domains, the user's actual behaviour is not performed online but in the natural world. This poses major questions about the monitoring of the actual user activity. If a video RS can easily access information such as watch time, the actual food consumed by a given user cannot be directly observed by the recommender.
Given that, several approaches were developed to monitor user consumptions and interests that are specific to food recommendations:

- **Explicit user feedback** is one of the most represented monitoring approaches and is not specific to food recommendation. In this setting, the user directly informs the system of his behaviour regarding the recommendation, either by just confirming the consumption or by rating it. This is commonly used in recipe recommendation, which is the closer food recommendation setting to other classical recommendation setups.

- Another commonly used approach is to rely on **questionnaires** on users' tastes. The idea is to build a user profile by using the answers of a given user to a set of questions on his or her eating habits. For example, [138] developed a food frequency questionnaire methodology to infer users' interests.

- **Food journaling** consists in asking the users to log their daily food consumption. By doing so, the system is able to follow the evolution of eating behaviour and infer users' preferences and tastes. Different forms of journaling exist, from ingredients writing [139] to meal photographing [140]. However, food journaling necessitates important user engagement. Moreover, it requires the system to be able to treat and interpret the collected data.

- In [141] the authors propose a method based on **image recognition**. By computing a similarity measure between meal photos, they are able to infer actual meal similarity. In [142], the authors use image recognition to deduce the food intake of users.

- Other methods based on **tracking consumption by using sensors** exist and have gained popularity in recent years with the development of the internet of things. For

example, in [143], the authors propose a framework to make food recommendations by considering the available food items in the fridge, which are detected by sensors.

To summarize, we can see that even if different approaches exist to monitor user behaviour in food RS, they are all engaging for the user. As such, experiments on real users necessitate motivated subjects, which is a prerequisite to the coaching approach.

4.1.3. Approaches

One of the main differences in existing healthy food RS is their approach to the recommendation task: what they recommend to users and how they communicate the recommendation to them. Given the recommendation objective (following daily intake guidelines, limiting obesity risk, managing diseases, etc.), diverse approaches exist and are explored in the literature. We present here the major ones.

4.1.3.1 Full diet recommendation

The first existing approach in the literature is to directly recommend a full diet. One can also refer to this type of recommendation as menu planning. The aim of this approach is to provide the user with a given planning over a given number of days to ensure the quality of the diet. Indeed, some works have shown the theoretical interest of people in healthy eating, but in practice, it remains a challenge for many people. One highlighted reason for that inconsistency is the difficulty for people to combine planning for a tasty and healthy menu and the rush of everyday life. Considering that setting, the goal of menu planning recommender systems is to lighten the organisation of a satisfactory meal plan by proposing suggestions that encompass the preferences and needs of their users.
For example, in [144], the author proposes a framework to make recommendations for a full week of food consumption to fulfil the dietary needs of elderly people and avoid malnutrition. To do so, the presented method relies on a hybrid recommendation approach, combining collaborative filtering and content-based filtering. The collaborative part accounts for the tastes and preferences of the users, while the content-based part focuses on other aspects of the considered recipes, such as dietary information or preparation complexity. These two aspects of recommendation are then combined using a constraint-satisfaction approach.
A more recent work on meal planning RS is the article of Caldeira et al. [145]. The

authors present a system whose goal is to recommend a set of meals depending on the available food in the user pantry. In contrast to [144], the system's focus is not on a given number of planned meals but on the maximum coverage of the accessible items. Thus, the size of the menu proposed by the system (i.e. the number of recommended meals) will depend on the user's pantry. The proposed approach provides the user with a list of meals that fulfil the requirements in proteins, carbohydrates and fat, use the available food items, and maximise a given notion of *harmony*, representing how well the food items of the recipe match with each other. To produce such recommendations, the authors propose a Pareto optimisation approach using a state-of-the-art algorithm. They show on real-world data that their method generates recommendations that ensure good pantry coverage and harmony while fulfilling dietary needs in carbohydrates, proteins and fat.

Although this is an interesting approach, allowing the system to consider the interaction between recommended items and dietary needs over a given period of time, meal planning recommendations can be too compelling for the user. Indeed, it necessitates a particular organisation and may limit the user in his/her choices.

4.1.3.2 Recipe recommendation

Another widely spread approach to the healthy food recommendation problem is recipe recommendation. The rapid growth of online recipe databases has essentially facilitated access to huge catalogues of recipes, accompanied by descriptions and a large amount of metadata. In this context, recipes can be seen as documents in the sense of textual information, as in news or articles. Thus, the recipe recommendation problem can be considered analogous to the extensively studied problem of news and articles recommendation. This, combined with the stakes around healthy eating and food recommendation, led to a great interest in the recommender system field for the task of healthy recipe recommendation.

Given the proximity between recipe recommendations and news recommendations, a classical approach is to make recommendations to a user navigating recipe websites. In this setting, recommendations can take the form of a "recommended for you" list. For example, the RS presented in [146] proposes to the user a list of recommended recipes, regarding both its interests and the healthiness of the considered recipes. The authors show that their developed system is able to make acceptable recommendations. However, they underlined the fact that healthy recipe recommendation is much more efficient when

addressed to users already interested in healthy eating. The same approach is used in [147]: the user is provided with a list of recommended recipes not evaluated yet.
On another side, the system presented in [148] proposes to substitute recipes with healthier ones. By taking advantage of the numerous recipes on Allrecipes.com [125], the system makes recommendations in the form of healthier alternatives to the recipe currently investigated by the user. By doing so, the presented work shows that it is possible to nudge a user towards healthier recipes using RS.

4.1.3.3 Joint food recommendation

When considering full diet or recipe recommendations, the user is expected to follow the recommendation strictly, and the recommendation is built on the inferred interests of the user. Another paradigm is for the system to build the recommendation jointly with the user. Indeed, by using the interaction with their users, this type of system can produce recommendations that are truly personalised and adapted to the current context of the user. Regarding whether the user or the system initiates the interaction, several methods for joint recommendation can be considered:

- **Recipe critiquing.** In the recipe critiquing setting, recommendations are full recipes, as in classic recipe recommendations. However, the user contributes to the final recommendation by proposing modifications of the recommended recipe. For example, in [149], the user is first provided with a primary recommendation taking in account healthiness, user preferences and food items indicated as available by the user. Then he or she may criticise the recommendation, for instance, informing the system that the recommended recipe is too spicy. The system then reevaluates the possible recommendations and makes a new recommendation that takes into account the expressed critics. A similar approach is presented in [150].

- **Food substitution recommendation.** Reciprocally to recipe critiquing, in the food substitution recommendation setting, the system recommendation is made in reaction to a choice of the user. In this approach, the system should be able to recommend to a user healthier substitutes for a given food item. This idea is investigated in [151], where the authors propose a system that recommends healthier substitutes extracted from a food knowledge graph representing the existing relations between food items. However, their approach is limited in that it does not take into account any notion of personal tastes. The focus of the study is on finding

functional substitutes that better fill the daily recommended intake in macro and micronutrients. Finding substitutes for food items has been of interest in recent years. For example, [152] or [121] present methods to extract substitutability from consumption data. Although not directly proposing food RS methods, these works evoke healthy recommendations as a future research goal.

- **Conversational recommenders.** Conversational RS, also referred to as chatbots, have largely gained in popularity in recent years. In particular, they have become increasingly used for health recommendation [153]. The idea of conversational RS is to let the user interact with the system in natural language, specifying his or her needs and expectations. Given the information furnished by the user, the system may either ask for clarification or make a recommendation. Answers given to the system questions are used to filter the recommendation space and to find an appropriate recommendation for the user. Like in critiquing, the user may react and ask for another recommendation. This methodology is used to nudge users towards healthy recipes in [154]. The presented system is able to interact with the user in natural language to propose healthy recipes and to explain the trade-offs between the different proposed alternatives. In addition to natural language, the work presented in [155] proposes to consider images and nutritional labels to further inform the user on the recommended recipes. The authors demonstrate the interest of such supplementary information when their system interacts with real users. Even the nutritional objectives can be chosen in interaction with the user. In [156], the user can specify his or her objectives, and the system can recommend recipes to fulfil these objectives.

4.1.4. Conclusion

To conclude, we can notice that the problem of healthy recommendations has increasingly gained in interest in the recent years. Many methods and algorithms exist, mainly derived from classical RS methods. The formalism of multi-stakeholder RS presented in Section 2.1.2 is surprisingly absent in the literature. The basic notion of healthiness taken into account in the literature is diverse, and no clear consensus appears on the best approach. Although an approach like full diet recommendations can guarantee an actual healthy diet, the impact of healthy recipe recommendations or joint food recommendations on the quality of a diet is questionable. Moreover, many works of literature underline the

importance of taking into account the long terms effects of healthy food RS [154, 147], but this issue remains largely unaddressed. Finally, suggestions to change eating habits in the long run are considered an essential yet unaddressed challenge of healthy food recommendation [157]. This opens room for a system focused on habit change and long-term, sustainable modification of the user's eating behaviour.

4.2. The problem of food recommendation as a coaching problem

We propose to address the problem of healthy food recommendation as a coaching problem defined in Chapter 3. Indeed, the presented framework appears to be well adapted to the challenges and specificity of the nutrition field. First, the coaching framework is designed to accompany users in a behaviour change process and is focused on the long term. In this sense, it appears to efficiently match one major stake of healthy food recommendation: inducing a shift in people's diets and eating habits. Second, the framework allows a high-level personalisation, which is identified as a key factor in food recommendation.
Moreover, the hypothesis of a coach knowing the actual value of the recommended items interacting with an uninformed user makes sense in nutrition. On the one hand, it exists nutritional scores that can be computed by an automated coach, as we discuss in 4.5. On the other hand, users may have an intuition of food healthiness, but the perceived healthiness can be misleading [158]. The scenario of repeated interaction is also well adapted to food choice, as it is a daily task, highly determined by one's personal habits [123].
Given that, our proposed model of the healthy food recommendation task is the following:

- **Items** are individual food items composing a meal. In this sense, we consider the recommendation task of suggesting intra-meal changes, that is, changes at an intermediate level between full meal recommendation and ingredient recommendation. Several considerations justify this choice. First, it meets the central idea of coaching, which is to build the recommendation on the user proposal: a complete meal substitution is a substantial change, and the user may feel less involved and considered in the recommendation process. Second, conversely to the ingredient-level recommendation, it can be applied in a broader context and not only in home cooking. In a refectory or a restaurant, for example, people can change their choice of starter or

dessert, but they cannot change the proposed recipes. Finally, substitutions at the ingredients level can be delicate, as some ingredients can be substituted in a given recipe but not in another one.

- **Users** are individual consumers willing to change their eating behaviour towards a healthier one. The coaching system models their habits through a choice vector over the set of food items and their acceptability of suggested substitutions through a substitutability matrix. The initialization of this matrix is discussed in Section 4.3.

- The **score function** is a nutritional score function measuring the healthiness of users' eating behaviours. It exists different nutritional scores, some of which are discussed in Section 4.5. For example, a possible one is the score of Rayner et al. discussed in Section 3.5.2, measuring individual food items' healthiness based on their macronutrient composition.

In the remainder of this chapter, we consider the presented problem of healthy eating coaching and investigate its real-world applicability. In particular, we explore some practical issues by conducting real-world experiments.

4.3. The *cold-start* problem

4.3.1. Introduction

When considering the application of the coaching method for real-world food recommendation, the first question that appears is a classical issue in RS literature, known as the *cold-start problem*. Cold-start is a problem due to the sparsity of data available for the recommendation. The key issue is to ensure a good quality of recommendation (i.e. recommendations ensuring a good level of performance when regarding the considered evaluation metrics) even with little data.
There are three types of cold-start problems regarding what type of data is too sparse: recommendations for new users, recommendations for new items, and recommendations for new items to new users [159].
In the case of coaching, the system is expected to learn from its interactions with the user a recommendation strategy to improve eating habits regarding a nutritional score function.

Moreover, the recommendations take the form of a suggestion of substitution for one item with another. All of the strategies investigated in Section 3.4 evaluate the possible substitutions using the given score function, the user characteristics or a combination of both. Thus, there are two possible sources of *cold-start*: on the one hand, insufficient data relative to items, leading to an impossibility of evaluating the score of a given item, and on the other hand, insufficient user data relative to the possible substitutions.

The choice of an adapted score function is open to discussion, and this question is the aim of section 4.5. However, when adopting the score developed by Rayner et al. and presented in 3.5.2, it can be easily computed from data of composition in macro and micronutrients of food items. Moreover, several databases exist furnishing such data on many food items. For example, the ciqual database [160] gathers data on the composition of 3 185 food items. So computing the score is not a limitation for the system and is not sensitive to the *cold-start* problem.

When considering the question of user data on possible food substitutions, we can notice that the problem is much more complex. Indeed, a user's reaction to a given recommendation depends on each user. Moreover, it is noticeable that the number of possible substitutions grows quadratically with the number of considered food items. As such, it appears totally unrealistic to learn from a single user the impact of each possible substitution. In addition, in the food domain, one can easily admit that not all substitutions are worth considering. For example, the suggestion to replace strawberries with meatballs seems completely uncanny, and learning the impact of such substitution on the user is of very little interest compared to plausible ones.

Considering the respective specificity of the two identified sources of *cold-start*, we propose in this section to focus on the user data and to investigate how consumption data can be used to address the user side of *cold-start*. Therefore, we examine the question of extracting, from observed consumption, substitutability values between food items.

4.3.2. Material and methods

We propose to use the methodology developed in [121] to extract values of substitutability between food items from consumption data. We apply the methodology as described in 3.5.3.3. We computed values of substitutability using the consumption data set INCA2. This data set contains individual 7-day food diaries of 2624 adults and 1455 children. We focus in this study on data for adults since children are not the target of a persuasive

food recommender system. A typical day of consumption, as reported in user diaries, is composed of three actual meals: breakfast, lunch, and dinner. The moments in between are denoted as snacking. To compute the values of substitutability, we followed the methodology of [121] and considered three sub-data-sets, corresponding to breakfast data, lunch and dinner data, and finally, lunch, dinner, and breakfast data altogether. Figure 4.2 presents a visualization of the obtained substitutability relationships for frequently consumed breakfast food.

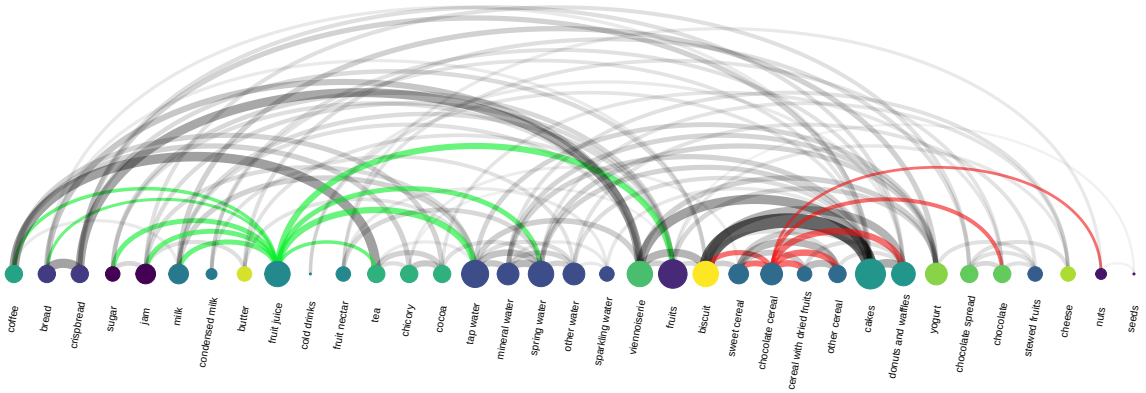

Figure 4.2: Arc diagram presenting the substitutability between the most consumed items for breakfast. Green edges show the substitutable items for fruit juice, and red edges show the substitutable items for chocolate cereal. Thickness of the edges indicate the strength of the substitutability relationship.

Regarding the obtained values of substitutability, a first qualitative analysis seems to indicate that they are realistic. We propose to test this hypothesis and the likelihood of the mined substitutability relationships.
To test the plausibility of substitutions proposed following the mined values, we designed an online task in the form of a Turing test. It consists in presenting substitutions to a pool of participants and asking them for each substitution if it has been proposed by a human being or by an artificial intelligence. The experiment was conducted and results were analysed by an internship student.

Participants. Volunteers were recruited to the online experiment via email. Emails were sent on a french public mailing list run by the French National Centre for Scientific Research (Information Relay in Cognitive Sciences, Paris, France, www.risc.cnrs.fr).
Participants were asked to be over 18 years of age and to be able to read and understand the French language properly to be included in the study. Each participant could not

participate more than once. Upon completion of the experiment, participants could enter a draw to win 15 €. A total of 255 participants participated in the study, and none of them reported having guessed the objective of the study.

Description of the online task. The experimental task consisted of three presentations of a series of 12 meals for which a suggestion of substitution was made. Proposals were made either by a professional dietitian or using the mined substitutability values. In the latter case, to test the relevance of the substitutability scoring system, suggestions either reflected substitutions with the highest substitutability score (expert algorithm), or substitutions with a low substitutability score (clumsy algorithm). All these suggestions concerned the same items of the 12 same meals. For each pair of meal + modified meal, participants had to answer ('yes' or 'no') to the following question: "some of these suggestions are made by an artificial intelligence, others by a dietitian, do you think this substitution was made by an artificial intelligence?". The supporting software was developed using the PC IBEX platform [161]. The chosen substitutability values used for the task are values computed on the lunch and dinner data set.

Data analysis. The dependent variable is the binary answer to the question on the emitter of the substitution recommendation (human/non-human). Binary logistic regression and resulting odds ratio were used to evaluate whether the answer was influenced by the actual type of emitter.

4.3.3. Results

A total of 255 participants participated in the study. None of them reported having guessed the objective of the study. Probabilities of recommendations for being judged as "made by non-human" are plotted in figure 4.3.

When comparing recommendations made by an expert dietitian and recommendations made following the mined substitutability values, we found that the probability that participants judge recommendations made by a human to be "made by non-humans" was low. On the contrary, the probability that participants judge as "made by non-humans" the recommendations made by following the substitutability score is significantly higher. That is, even if substitutability values seemed qualitatively plausible, participants were able to make a distinction between recommendations based on it and real dietitian recommendations.

However, regarding the results obtained by using high substitutability values, and low

substitutability values, it is noticeable that recommendations made from high substitutability values are more frequently judged as "made by humans". The difference is far from the one existing with human-made recommendations, yet statistically significant. In other terms, the recommendations made from the high substitutability values appear to the participants as comparatively closer to recommendations made by a dietitian than recommendations derived from low substitutability values.

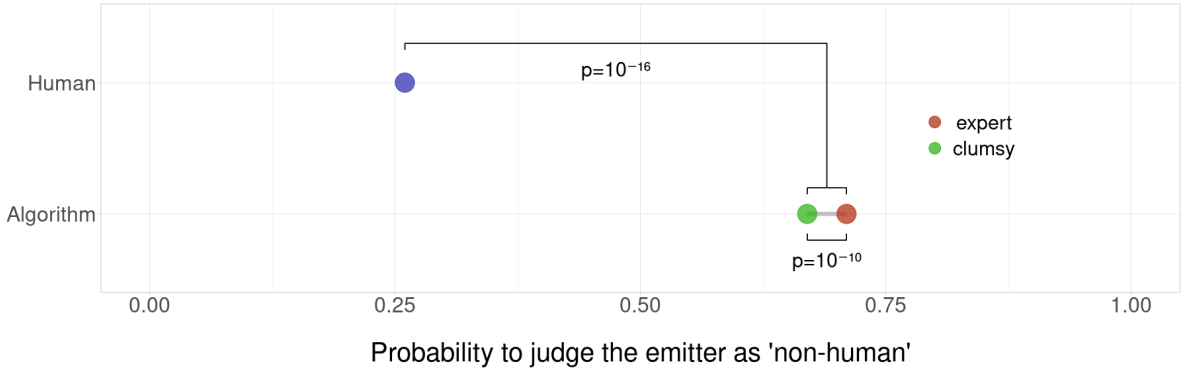

Figure 4.3: Probability to judge recommendation as "made by a non-human" as a function of type of emitter. Probabilities and p-values are obtained via a binary logistic regression model.

4.3.4. Discussion

The presented results indicate that the recommendation computed from consumption data were, in the majority, recognized as emitted by a non-human agent. This indicates that there still exists a margin of progress in the ability of the developed method to mimic human-emitted substitutions from the available consumption databases. However, it appears that the higher the substitutability score, the higher the probability of the substitution being judged as human-emitted. Thus, the presented method can be seen as a perfectible proxy of the substitution's plausibility. A potential extension that could lead to more plausible recommendations would be to take into account contextual information influencing food choices, such as time of the day or previously consumed meals. However, it is necessary to be careful in these results' interpretation. Indeed, we assumed that the plausibility of a substitution is associated to higher acceptability. This is not necessarily the case, and very plausible suggestions could be difficult to actually implement for the user to whom they are recommended. Thus, in addition to substitutability values, it is

important to take in account other factors for the acceptability of the recommendations, such as for example, the presentation mode of the suggestion. (See Section 4.4.)
Nevertheless, the investigated method for substitution mining appears as a primary information source when interacting with new users. In the particular case of cold-start, the gathered information about substitutability relationship between food items can be used to initialize the recommendation algorithm, avoiding as much as possible "non-sense" recommendations that can lead to very poor efficiency of the system. Moreover, it is known that users generally use health apps for a limited period of time [162], and we can suppose that limiting the "non-sense" recommendations is key in improving user retention and avoiding early dropout of the recommender system.

4.4. Human-computer interaction : investigating the coaching interface

4.4.1. Introduction

As stated in Section 4.3.4, numerous factors may influence the acceptability of a healthy food substitution suggestion. In particular, the human-computer interaction design can lead to great differences in the user response to a given recommendation. Given the objective of coaching, which is focused on user behaviour change, the acceptability of the recommendation is a critical question. Moreover, as the healthiness of the recommended food and user's tastes may be antagonistic, maximizing the acceptability of recommendation by every possible means appears fundamental.
The question of recommendation acceptability is key in the recommender system literature. Beyond the actual computing of the recommendation, the way the system interacts with the user and presents the recommendation are also research questions investigated in the field. For example, much research work focused on critiquing [163] as a way to propose more acceptable recommendations to users. Critiquing is mainly used in recommendation scenarios where items are not frequently consumed, like when recommending cars or laptops. In these types of scenarios, the task of building a meaningful user profile from consumption data may be complicated, if not impossible. Thus, the critiquing phase makes it possible to elicit user preferences. However, critiquing can also be used in combination with classical recommendation methods to improve the accuracy of the rec-

ommendation and the trust in the system, as in [164]. In healthy food recommendation, critiquing has also been used to improve the acceptability of the recommendation [149]. The effect of critiquing on the user's potential trust in the system appears to be very meaningful in a scenario like coaching, where the announced goal is to provide the user with healthier eating habits. Therefore, we are interested in investigating the possible modes of recommendation that could be applied in a coaching interaction. Moreover, we aim to find a recommendation mode that maximizes the acceptability of recommendation in a coaching context. Indeed, as denoted in section 4.1.3.3, coaching can be considered as a form of inverse critiquing, where the system critiques the proposal of the user. As such, the coaching framework itself already partly defines the interaction scenario. Our question then is to what extent different modes of recommendation can be adapted to it, and what is their effect on the acceptability of recommendations. The experimental protocol was designed in collaboration with experts and an internship student. The experiment was conducted, and the results were analyzed by the internship student.

4.4.2. Material and methods

In order to investigate the effect of how the recommendation should be provided to the user, we proposed to test different recommendation modalities in a real-world coaching scenario. Thus, a mobile healthy food coaching interface was developed for the purpose of the experiment. The aim of this interface is to let the user enter a meal and to provide him or her with a suggestion for substitution. Participants were asked to declare the meal they intended to eat one day in advance. So they received the recommendation of the coach one day in advance in order to make it as easy as possible for them to implement it.

The principle of the interaction follows the scenario of coaching described in Chapter 3 :

1. The participant declares the meal they intend to eat the next day to the coach

2. The virtual coach suggests a substitution into this meal. The modality of the recommendation is the aim of the study, and three modalities, described below, were tested

3. The participant accepts or refuses the suggestion

4. If a suggestion is accepted, the participant commits to implement the recommendation and to certify it by sending a picture of his or her meal.

Our hypothesis on recommendation modalities is that the more the user of the system feels involved in the recommendation interaction, the more acceptable the possible suggestions of substitutions. This hypothesis is based on a known effect in human psychology: the higher participants are engaged in a task, the higher they value the results of this task [165]. Here we hypothesise that a higher perceived value may lead to a higher acceptability of the recommendation.

To test this hypothesis, we designed three recommendation modalities with different levels of involvement. In the *first modality*, we propose to test a critiquing scenario. That is, when the user receives the suggestion, he or she may emit critiques about this suggestion to indicate that he or she is not satisfied with one or more features of the suggestion. For example, the user may indicate that the suggestion is "too salty". The system then takes into account the user response and makes a new suggestion. This is the basic form of critiquing recommendation in the RS literature. In a *second modality*, we propose to test a scenario where the user is involved in the recommendation and able to specify precisely his or her preferences lie in the critiquing scenario. But to test the importance of the interaction between the coach and the user, we consider here a modality where the user first describes his/her preferences and then is provided with a unique recommendation, taking into account the expressed preferences. In this modality, the user is involved in the recommendation, but the recommendation is not constructed gradually via an exchange between the system and the user as in critiquing. Finally, in a *third modality*, the user cannot specify any preferences and is provided directly with the recommendation.

For each declared meal, the system generates four suggestions of substitutions based on the mined values of substitutability for lunch and dinner, as presented in Section 4.3, and the Rayner et. al nutritional score. The only considered substitutions were substitutions that allow a score gain. Then four substitutions with the highest substitutability value were chosen.

The resulting three tested modalities are the following :

- **Modality A:** all four options are presented simultaneously, and the user can choose either one or nothing.
- **Modality B:** identified suggestions are presented one by one. At each time, the user can refuse the suggestion. If doing so, he or she is asked to justify the refusal

by critiquing the suggestion. The next suggestion is the one that better fits the critique in the not yet presented ones. If the last suggestion is not accepted, no option is chosen.

- **Modality C:** the coach asks the user for his or her preferences regarding a given list of components. Then it makes a single suggestion taken from the four generated suggestions, which best match the announced criteria. The user can either accept or refuse.

As the aim of this study is to test to what extent interaction impacts the acceptability of suggested substitutions, we want to limit as much as possible the effects due to the intrinsic quality of the substitutions. To do so, we proposed to pre-compute the possible suggestions, regardless of the chosen modality. Then only the presentation of the results depends on the modality. In other words, recommendations from modality B or C, which take into account the preferences of the user, are not better suited or much more personalized than recommendations from modality A. The user only has an illusion of control over the recommended items.

Participants. Based on similar studies, we estimated that 30 participants were needed for this study. Considering a dropout and non-completion rate of the experiment of 50%, to obtain approximately 30 complete and exploitable responses, 60 candidates were recruited. The recruitment was done via an online form distributed via a public mailing list run by the French National Centre for Scientific Research (Information Relay in Cognitive Sciences, Paris, France, www.risc.cnrs.fr). The inclusion criteria were being over 18 years old, not being on a diet, and owning a smartphone.

Ethics approval. The study was conducted according to the Helsinki declaration guidelines, and all procedures were approved by the Ethics Committee of Université Paris-Saclay (decision CER-Paris-Saclay-2021-055). Written informed consent was obtained from all participants. Access to the General Data Protection Regulation (GDPR) is permanently available from the application interface. Participation in the experiment was compensated by a gift voucher worth 50€.

Conduct of the experiment. The study used a within-subject design to test two meal conditions. The experiment was conducted for three weeks, spanning June and July 2021. Each participant interacts two times a week with the coach. On Monday and Thursday, participants are asked to declare their intended meals for Tuesday and Friday

evenings, respectively. Thus each participant interacts with the coach six individual times. Each week, a different modality of recommendation is used. Therefore, every participant observes twice the three individual modalities. To avoid possible effects of the order of presentation, participants received the different modalities in a randomized order.

Measured parameters. At the beginning of the experiment, the volunteers filled out a questionnaire indicating their age, sex and BMI. For each recommendation session, the acceptance or refusal data were recorded, as well as the constituents of the meals filled in by the volunteers and the elements suggested by the coach.

Statistical Analysis. In order to explain the influence of the mode by which the recommendation is given on the probability of acceptance, a binary logistic regression analysis has been implemented. The statistical model used is therefore described as follows:

$$P(accept) \sim Sex + Modality + Sessionnumber + age + BMI$$

To represent the probabilities of acceptance, odds ratios were computed from the logistic regression model.

4.4.3. Results

Of the 60 candidates initially enrolled in the experiment, the results of 27 of them were complete and exploitable for analysis (yielding a total of 162 meals). The final sample was composed of 20 women and 7 men. The average age was 37.5 ± 15.2 years. The average BMI was 22.2 ± 4.1 with two overweight individuals and two others moderately obese. Of the 162 interaction outcomes between the participants and the coach, we observed that 74 of them resulted in the acceptance of a recommendation, reflecting an overall average acceptability of 46%.

Analysis of the odds ratios corresponding to the different factors that influence the acceptability is presented in Figure 4.4. Only modality B ($OR = 3.168$, $CI95\% = 1.688 - 6.061$, $p < 0.0004$) was associated with an odds ratio exceeding the significance threshold. A tendency was noted for the effect of age (p=0.08), indicating that younger participants may have a higher propensity to accept recommendations. Sex (p=0.14) and BMI (p=0.32) did not have a significant effect on acceptability.

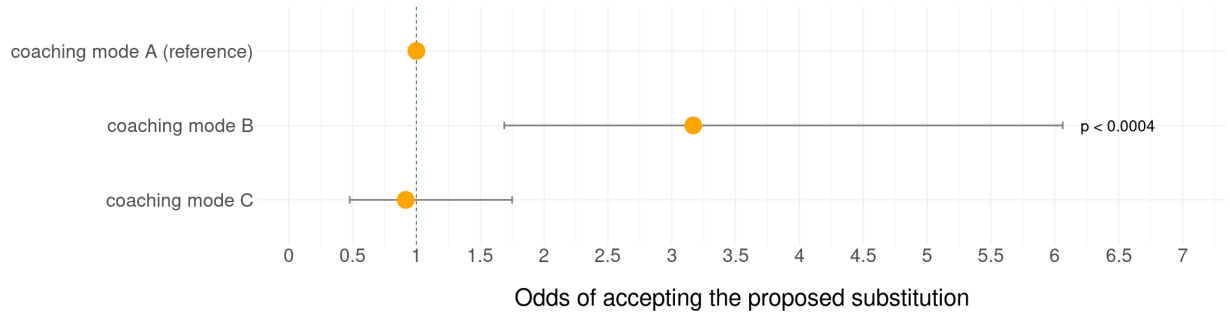

Figure 4.4: Effect of different coaching modalities on the probability of accepting the recommendation. Results are presented as odds-ratios and 95% confidence intervals. The P-value presented here is the result of the binomial logistic regression analysis.

4.4.4. Discussion

We proposed to test the effect of interaction design on the acceptability of suggested food item substitutions in a meal. The conducted experiment shows the interest of modality B, that is, an interaction scenario based on recommendation critiquing, over two other tested modalities (raw recommendation list and pre-recommendation preference elicitation). In other words, compared to a recommendation where all suggestions are proposed to the user, the gradual elaboration of the recommendation in interaction with the system seems to favour the acceptability of the suggestions. Moreover, the control of the user on the recommendation has not to be real, and the illusion of control already induces effects on acceptability. Indeed in the presented experiments, modality A and modality B are based on the same four suggestions. Thus it is not the actual personalization of recommendations that induces the observed differences but possibly the involvement of the user in the recommendation elaboration. Indeed studies in psychology or behavioural economics have established that the more we engage in a task, the more we are attached to the results and the greater the value of this result is for us.

Contrastingly, when investigating the results of modality C on acceptability, we found no significant difference with modality A and significantly worse acceptability compared to modality B. However, the user is asked to inform the system about his or her preferences to generate the recommendation, as in modality B. Two main reasons can be considered to explain these results. First, a coach based on modality C proposes a unique recommendation for each interaction. On the contrary, modalities A and B allow the user to explore at most four different suggestions. Therefore a user has more chances to find an accept-

able suggestion as the recommendation sample-size increases. Second, the engagement of the user may be more important in modality B compared to modality C, as the user can observe the evolution of the suggestion generated by the expressed critiques. This makes the suggestion both more engaging and more explainable for the user, as he or she can observe the reasons leading to it. And explainability is known as a determinant of acceptability of recommendation when interacting with recommender systems [166].
Nevertheless, it is important to notice that the observed results should not be generalized, as the profile of the participants remains not representative, being constituted mainly by young women. Among the tested interaction modalities, critiquing seems to perform the best when considering the acceptability of the suggested food substitutions. However, the number of participants and the design of the study did not allow us to identify different profiles. Thus we cannot exclude that the best interaction modality depends on the considered user group.

4.5. Choice of a nutritional score

As presented in Chapter 3, coaching is based on an evaluation of the task at hand through a score function. That score function is supposed to be known by the coach but not by the user. Then the objective of the coach is to train the user so that he or she maximises the value of the score associated with his or her behaviour.
It exists several score functions in the food domain to evaluate the healthiness of food items or diets. We have evoked in the previous chapters the score designed by Rayner et al. to assess the nutritional quality of individual food items, but the scores' objectives and domains of application are diverse. Two main paradigms exist, depending on what is to be evaluated: diet healthiness scores and food items healthiness scores. In the following, we review the major contributions of the literature on those two types of scores and discuss their interest in a healthy food recommendation coaching context.

4.5.1. Evaluating healthiness of food items

As their name suggests, food items' healthiness scores are based on a categorisation of each individual food item. In this setting, healthiness is a feature relative to the item. One can find diverse such scores in the literature [167, 120, 168]. They are generally computed from the composition in specific nutrients of each food item. Such approaches are known

as nutrient profiling [169]. Several models exist for nutrient profiling, depending on the considered nutrients. Some are based on nutrients to limit or avoid, others on nutrients to favour, or a combination of both.

This type of score is mainly used in public policies to regulate marketing or health advertising or to inform the consumer. For example, in France, the five-colour nutrition label of Nutri-Score [170] has been largely adopted as a standard for nutritional labelling. However, another possible application of such scores can be to help the consumer to choose healthier food and to educate them by providing nutritional information [171]. In the context of coaching, this is of great interest. Indeed, the aim of coaching is to help a user (or consumer) to change his or her dietary habits towards healthier ones, and food-based nutritional indices are making it possible for the coaching system to evaluate the healthiness of user choices. Moreover, studies have shown the interest of some nutrient profiling derived indices to promote healthy diets [167].

When considering the context of automated coaching, this type of nutritional score appears very well adapted to the developed framework. As the nutritional value of an item is computed only from its composition, the scores can be pre-computed, and the system does not have to gather extra contextual information to be able to make a recommendation. In addition, the framework was proposed with a given value attached to each item, which is the case with this kind of score. However, when considering the long-term objective of coaching, limitations in the applicability of food-based nutritional scores appear. Indeed, consuming items with high values regarding some nutrient-based indices has been proven to be associated with a healthier diet. Nevertheless, these are not tools designed to measure the overall quality of diets. Evaluating the healthiness of a diet is a complex question involving diverse and interdependent factors. A simple mean food item value does not reflect this complexity, and by trying to maximise it, a system relying only on individual food items' nutritional values would not be fully efficient in promoting healthy eating habits. Diet has to be considered as a whole when evaluating healthiness.

4.5.2. Dietary quality indices

Conversely to food-based nutritional indices, dietary indices focus on evaluating whole diet quality. A wide variety of such indices exist, depending on the country and the evaluation methods. Indeed, when investigating the literature, one can categorize diet quality indices in two main approaches: on one hand, indices based on national food-

based dietary guidelines and on the other hand, indices based on nutrient intake.

Indices based on dietary guidelines measure the accordance of a given diet with the considered guidelines. Generally, dietary guidelines consider multiple components and focus on consumption levels of a given set of food groups, such as vegetables, meat and so on. They can also integrate some information about nutrients. For example, the index presented in [172] integrates a measure for salt consumption. As nutritional guidelines vary across countries, it exists many local indices [172, 173, 174, 175]. Also exists a more general index, the healthy diet indicator (HDI) [176], based on the world health organization guidelines. Besides indices based on proper dietary guidelines, were developed indices to measure the accordance to the "Mediterranean diet" which is known to be significantly associated with reduced risks of cardiovascular diseases and several forms of cancer. An example of such a score is the MedDietScore [177]. The vast majority of these indices have been significantly associated with positive health outcomes. For example, high HDI values are associated with significantly lower overall mortality [176]. In France, the value of the national index has also proven to be highly correlated with obesity and overweight [178], while the guidelines used to build the score have proven to be associated with better nutrient intake [179]. However, this type of index suffers from methodological limitations: the choice of the considered parameters and their relative weight in the final index has to be discussed [180].

The other approach for computing indices to measure the quality of a diet is to base the measure on the accordance of the considered diet to recommended upper and lower intakes of nutrients. The benefits of such an approach are two-fold: on the one hand, this type of index is more easily adaptable to different geographical areas. On the other hand, there is a large body of literature validating the importance of nutrient intake in health outcomes. Some indices have been developed following this idea. For example, the mean adequacy ratio index has been used in [181] to measure diet quality. More recently, the authors of [182] have proposed an index based on the probability of adequate nutrient intake.

In the context of healthy food recommendation by coaching, these two approaches are of interest: by contrast to food item indices, dietary quality indices are constructed to measure the quality of a diet. Thus making recommendations that consider this type of index as the score function is expected to lead the user towards a healthier diet. In that sense, these indices seem to match the long-term objective of coaching. However, from an operational point of view, the use of this type of index presents some challenges.

Indeed, all of the considered diet quality indices from both approaches are computed over a certain time period. It can be daily [176], weekly [178, 182] or even monthly [177]. This raises questions about how a coaching system can consider these indices: to compute the nutritional interest of a given suggested substitution, the system computes the score gain permitted by this substitution. In the case of indices computed over a certain period of time, the system has no access to individual values of substitutions. A way to bypass this limitation is to compute the given index on a sliding time window. But in that case, a given substitution may not have the same value depending on the previously consumed items. For example, recommending to substitute chicken for fish may be highly interesting if the considered user has not recently eaten fish but is far less valuable if he or she has recently eaten a lot of fish. This example lets appear a key concept that has to be considered when using this type of index: the *context* of the recommendation. In other words, making coaching recommendations to improve the whole diet of a user necessitates adapting the coaching framework so as to consider contextual information.

4.6. Conclusion

In this chapter, we proposed to apply the coaching framework presented in Chapter 3 to the problem of healthy food recommendations and long-term improvement of users eating habits. We reviewed the state-of-the-art solutions in healthy food recommender systems and underlined the lack of long-term consideration in the literature.
In order to test the applicability of coaching to healthy food recommendation in a real-world scenario, we designed two experiments. The first tested the capacity to extract meaningful substitutability data from consumption data to overcome the cold-start problem. The second investigated the design of the interaction scenario in a coaching setting and its impact on the acceptability of recommendations.
We proposed a method based on the work of [121] and showed that it could be used to identify plausible substitution from consumption data. However, the approach is still perfectible. Moreover, we highlighted the importance of user involvement in the recommendation process to further enhance the acceptability of the recommendations. Finally, we discussed the notion of nutritional score and presented the two main paradigms: food item scores and full diet scores. We emphasize the importance of considering the healthiness of a whole diet in recommendation and elicit the importance of integrating contextual information in the coaching framework.

5. Contextual coaching

As evoked in Section 4.5.2, the evaluation of food consumption in a healthy food recommendation scenario may benefit from considering food quality indices, as they are more appropriate to measure a diet's quality. This poses the question of the recommendation context and how it should be taken into account in a coaching interaction. In the following chapter, we propose to explore how coaching systems can benefit from context and how recommendations can take this into account. In Section 5.1, we define the notion of context and discuss its importance in the case of coaching. Section 5.2 presents a review of the related literature. In Section 5.3, we study the best possible behaviour for a given user in the case of contextual coaching, while Section 5.4 investigates the question of the coach recommendation strategy in such a case. Finally, in Section 5.5, we discuss the proposed formalism and present future research directions.

5.1. Introduction

5.1.1. The notion of context

Context is a hard-to-define concept, but that can strongly influence decision-making, particularly human decision-making. In [183], Suchman explores the notion of "situated action". For her, every human action is performed in a given situation and depends on this so-called situation, hence the "situated action" concept. In other words, she considers an action inextricably linked to the situation in which it occurs. The notion of *situation* as defined is broad as it encompasses any particular concrete circumstance, including material and sociological conditions. The situation underlying the action is called the *context* of that action. In that sense, the context in which an action is taken influence decision-making: Action is considered context-dependent.

Nevertheless, context does not only influence action. It is also known to affect learning. In [184], the author shows the existence of "contextual association,", which refers to connections in memory between what is learned and the context in which the learning

occurs. That is, it is easier to remember a given information in a context similar to the one in which the learning occurred.

Hence the importance of taking context into account when dealing with human decision-making. However, the exact definition of what constitutes context is still debated among researchers. Indeed, when considering the approach of Suchman, the context is the situation in which the action is performed, in other words, the external environment in which it takes place. However, context can also refer to the internal mental state of the decision maker [185]. Moreover, context is inherently subject to constant changes as actions may influence the environment and so the context of future actions.

When considering human-computer interaction, a widely adopted definition of context is the one proposed by Dey in [186]: "Context is any information that can be used to characterise the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves". Although considering a comprehensive range of possibilities, this definition points out the essential notion of relevance. Indeed, if context can be constituted of a tremendous number of factors, only some are relevant considering the situation at hand.

As such, many research fields, including recommender systems, have considered and studied the notion of context.

5.1.2. Context in the coaching framework

When considering the coaching interaction described in Chapter 3, taking the context into account appears essential. Indeed the essence of the coaching framework is to impact the user decision making: the system aims to make recommendations to accompany the user towards a new behaviour. Nevertheless, as we have seen, the user decisions, thus behaviour, may be impacted by the context.

Moreover, the actual evaluation of the user behaviour by the coaching system may also take into account the context. Indeed, in coaching, the system's objective is to lead the user towards behaviour that maximizes a given score function. However, this score function may consider contextual data in the evaluation. For example, as seen in Section 4.5, evaluation indices exist in the nutrition domain that depends on the items' consumption over a certain time period. When considering such an index, the value of a given item depends on the other consumption over the time period, that is, on its context of con-

sumption. To illustrate this, let us consider the following situation. When consuming fish, the nutritional interest of the consumption is not the same if it is the first consumption of the last seven days or the tenth. Indeed, if eating fish is essential for the intake of omega-3, eating fish too frequently is not recommended. This is true for nutrition but can also be observed in other domains. For example, using an underarm serve can surprise the opponent and lead to a winning point when playing tennis. However, using it too frequently, the player will lose the surprise effect and may eventually lose the point. In other words, the "value" of such a shot for a tennis player depends on the sequence of previous shots. These examples underline that bringing context into coaching is essential, not only from the user side but also from the coach side. The interest of a given action under a specific objective may depend on the context that leads to the action. In the rest of this chapter, we will focus on the questions raised by the context from the coach side and on the problem of incorporating context in the evaluation of user behaviour.

The question that emerges then is how to make recommendations that consider a contextual evaluation function of the user behaviour in a coaching interaction. Actually, a coaching system's efficiency in accompanying a user towards a behaviour that maximizes a contextual score function depends on the user's capacity to implement this behaviour. Indeed, coaching systems are, in essence, building their recommendation on the observed user behaviour and relying on their user learning ability to maximize their long-term impact. In other words, a contextual coaching system's efficiency depends on the user's capacity to perceive the context and behave adequately regarding it.

5.1.3. The problem of incomplete information

In a setting of contextual coaching, the key question lies in the capacity of the user to perceive the context and implement adapted responses. Indeed, if, on the one hand, the observability of the context for the system may be incomplete, here we assume that the system gathers all the necessary contextual data to compute the score function associated with the coaching problem. On the other hand, the user representation of the context may be limited. A given user may not perceive all the contextual features of a situation that are, however, of interest in evaluating the situation. That is, a user may not be able to represent all the potential contexts and the best action associated with each of those according to the score function. As an example, he or she may consider two contexts that are evaluated differently by the coach identical. We call *user representation space* the

set of all the distinct contexts that the user can perceive. At the end of the day, we can identify two major cases:

- The user representation space is large enough to encompass all the contextual information needed to compute the score function. In this case, for each situation the user encounters, he or she can potentially learn the perfect action to perform to maximize the score function. In other words, the score function is *representable* in the user space and, therefore, can theoretically be learned by the user.

- The user representation space is too limited to encompass all the contextual information needed to compute the score function. In this case, the user faces the *incomplete information* problem. Even a user that would try his/her best to maximize the score cannot behave in a way that guarantees the actual maximum of the score function, as he/she cannot represent all of the required information. In other words, when facing a given situation, the user does not have access to enough information to determine the best action regarding the score function.

These two cases highlight the relative importance of the user representation capacity compared to the dimension of the score function considered, that is, the space of distinct contexts considered by the score function. Moreover, they pose the question of how to take into account the possible difference between user representation capacity and score dimension in a coaching scenario. What objective should the coach pursue, and what should it recommend regarding this objective? We identified three major research questions arising from the integration of contextual evaluation in coaching, which we present below. First, given a user representation capacity, a coaching system should be able to compute a target behaviour for that user, hence:

- ***Q1:*** *Given a user representation space and a contextual score function, what representable behaviours maximize the score function? What should be the target behaviour of the user?*

Second, when a coaching system has identified the target behaviour for the user, it should be able to make recommendations that lead the user towards it:

- ***Q2:*** *What recommendation strategy should a coach apply to accompany a given user towards a target behaviour?*

Finally, computing a target behaviour necessitates knowing the representation space of the coached user:

- ***Q3:*** *How a coaching system can estimate, from its interactions with a given user, his/her capacity of representation?*

In the remainder of this chapter, we will focus on *Q1* and *Q2*. Possible research directions for *Q3* are discussed in Section 5.5.

5.2. Related works

5.2.1. Context in recommender systems

If the context is naturally understood by human beings and has even a role of disambiguation in natural language and human-human interactions, it is much more difficult to consider the context in human-computer interaction. Indeed, the scope of the interaction is inherently much more limited, as a computer only gathers a predefined set of descriptive features. Consequently, computers cannot directly take advantage of the context when interacting with humans, as humans would. Nevertheless, context is known to be an important determinant of human decision-making. Thus its importance in human-computer interaction is critical when considering recommender systems that aim to influence decision-making or even predict decisions. Therefore, the context has been studied from the early 2000s in the RS literature. In particular, a whole part of the literature in the domain has been interested in *Context aware recommender systems* (CARS). The two main questions addressed by CARS are, on the one hand, the definition and modelling of context and, on the other hand, the integration of contextual information in recommendation algorithms.
According to [187], there exist two main views of context that are respectively inherited from *positivist* and *phenomenologist* theories:

- *Representational view*: The representational view of context is based on the question of encoding and representing context. In this view, context is considered a supplementary form of information that has to be gathered by the system. As such, context is also considered delineable and stable: what is and is not context is defined

before any human-computer interaction. More important, context and activity are distinct. An activity takes place in a context but does not form part of it.

- *Interactional view*: The interactional view of context is based on the activity dynamics. Context is not seen as distinct from the activity at hand. It is rather both influencing it and resulting from it. Context is considered permanently evolving, and its relevance may change along the interaction.

Even though the vast majority of works in CARS literature focus on the representational view of context, the interactional view sheds important light on the interdependence and inter-definition between context and activity. The classical approach of CARS is to consider a set of *contextual factors* that define the current context in which a recommendation is made. These factors can encompass different dimensions of context and thus may have various structures. As depicted in Figure 5.1, the authors of [188] propose to consider contextual factors along two dimensions. The first one is observability, which describes to what extent the contextual factors are explicitly known by the recommender systems. The second is how contextual factors change over time: they can be static and remain stable over time as in the representational view, whether as dynamic as in the interactional view, and have both their structure and/or relevance changing over time.

How Contextual Factors Change	Knowledge of the RS about the Contextual Factors		
	Fully Observable	Partially Observable	Unobservable
Static	Everything Known about Context	Partial and Static Context Knowledge	Latent Knowledge of Context
Dynamic	Context Relevance Is Dynamic	Partial and Dynamic Context Knowledge	Nothing Is Known about Context

Figure 5.1: Dimensions of contextual information for CARS [188]

Regarding this classification, we can notice that the considered problem of context in coaching lies in the dynamic observable case: the coach is informed about context, but its relevance for the recommendation evolves over time. Moreover, in our case, the context does impact not only the user behaviour but also the intrinsic quality of the recommended items from the system's point of view. Thus in our case, context cannot be described under the representational view, conversely to most works in CARS.
The question of *incomplete information* has been studied in CARS literature. However, in contrast to our question, the focus is on the recommender system and its incomplete

perception of context. As such, the existing works propose methods to infer contextual information from user feedback efficiently. For example, in [189], the authors propose a parsing method that extracts from user reviews the relevant contextual factors using natural language processing methods. Similarly, the authors of [190] use review mining to infer contextual information via supervised topic modelling.
As our work is interested in contextual recommendation, the CARS literature gives us a framework and an operational definition of context. Nonetheless, it is of little help when considering the actual recommendation problem faced by a coaching system based on a contextual score function.

5.2.2. Sequence-aware recommender systems

A particular form of context-aware recommender systems investigated in the literature are sequence-aware recommender systems. As described in [191], these consider the sequence of user interaction logs when computing recommendations. The idea behind sequence-aware recommender systems is that the ordered sequence of past user choices/decisions is informative for the recommendation, in the same way as ratings of items. Thus several methods were developed, that differ from classic recommender systems approaches, in order to take into account the importance of the sequence.
According to [192], sequence-aware recommendation tasks can be categorized into four main problems:

- **Context adaptation** aim is to adapt the recommendation to the context of the user, by considering the past action sequence as the context. In this setting, the context is definitely considered through the *interactional view*. However, the context is seen as an additional data source leveraged to inform the system of the user's preferences and expected choices. By contrast, in the problem of contextual coaching, we consider the problem of the contextual evaluation of recommendations from the system viewpoint.

- **Trend detection** focuses on the problem of inferring from the sequential data trends in the evolution of users' behaviours or preferences.

- **Repeated recommendation** is defined as the problem of identifying patterns in the users' behaviour, so as to make adapted recommendations given these repeated patterns.

- **Consideration of order constraints and sequential patterns** encompasses the questions about the ordering of the recommended items. Indeed, may exist relations between items relative to their order of consumption. A simple example is the one of TV series: the interest in recommending a given episode probably depends on the previously watched episodes. This is particularly relevant in the field of path recommendation where some items are prerequisites for others, as discussed in 6.2.2.

Thus, although of interest to our work, the literature on sequence-aware recommender systems does not directly address the problem investigated in this chapter. On the one hand, context adaptation is mainly focused on the extraction of users' information from sequential data. On the other hand, the hierarchical approach of path recommendation may be of interest for coaching but does not meet our focus on the contextual evaluation of user behaviour. Moreover, when considering our first research question, the problem seems more closely related to the research around knowledge representation and distillation than to the research on sequential recommendation.

5.2.3. Knowledge distillation

When considering our first research question of finding the best target behaviour for a given user depending on his/her representation capacity, one can notice that it belongs more to the domain of knowledge representation than the domain of recommendation. Indeed, the core question is one of representing complex behaviour in a limited representation space. This can be seen as a problem of model compression as described in [73]: the objective is for the coaching system to compress its knowledge into a smaller representation space (the user representation space) so as to find a policy that associates an action that maximizes the considered contextual score function to each representable state for the user. Based on the idea of model compression, Hinton [72] proposes a method known as "knowledge distillation" whose aim is to transfer the knowledge from large models (teacher models) with large representation spaces to a smaller model (student model). However, even if the underlying concept is of interest to our work, most of the existing research is focused on distilling knowledge from large neural networks to smaller ones [193]. In practical terms, most of the works take advantage of the learning and generalization capacity of a classifier to learn an accurate decision function from the outputs of the teacher model. Given our model of user knowledge, it is not straightforward to apply methods from knowledge distillation to our question. Nevertheless, training a user to

maximize a given contextual score in a limited representation space can be conceptually seen as a form of knowledge distillation.

5.3. Best representable behaviour

Our first focus is on the question of the representable user behaviours that maximize a given contextual score function. In this section, we first propose a general formalization of this problem. We then propose to study this problem in the particular case of historical context, as this type of context is particularly relevant to the coaching setting.

5.3.1. General problem formalization

Let us model the general problem of finding a behaviour that maximizes a contextual score function. Consider a score function S, which is associated with an item set I. Here we focus on the problem of choosing one unique item $i \in I$ at each step $t \in T$. We consider that each item $i \in I$ is associated with a value of S that depends on a so-called context, denoted H. The score function S represents the interest, regarding the objective of the coaching interaction, of choosing item i in context H. We denote $\mathcal{H}$, the set of all possible contexts. We do not focus for now on the actual form of H. It can represent all kinds of information that is relevant to the evaluation of an item choice. So the score function S associates to each couple (i, H) a value representing the interest of choosing item i in context H, hence the notation:

$$S : \begin{cases} I \times \mathcal{H} \to \mathbb{R} \\ (i, H) \mapsto S(i, H) \end{cases} \tag{5.1}$$

Given such a score function, we consider a user U whose task is to choose at each step $t \in T$ an item $i \in I$ such as to maximise the mean value of the score function S over T steps. As stated in section 5.1.3, we propose to consider for the user U a representation space denoted R_{U}, that model the set of all the contexts U is able to perceive. Formally, "perceiving" a context for U can be seen as a mapping $\mathcal{M}$ from $\mathcal{H}$ to R_{U}. We note:

$$\mathcal{M} : \begin{cases} \mathcal{H} \to R_{\mathsf{U}} \\ H \mapsto \mathcal{M}(H) = \widehat{H} \end{cases} \tag{5.2}$$

with $\widehat{H}$ the context perceived by U. In the following, we consider both $\mathcal{H}$ and R_{U} as finite. Hence, one can notice that the cardinality $|R_{\mathsf{U}}|$ is the number of contexts that U can represent.

Now consider the behaviour of U. We propose to model it by a decision function denoted D_{U}. We assume that the behaviour depends on the context perceived by U. Indeed, as denoted in the recommender system literature, context is known to influence decision-making. Moreover, we assume that the contextual data that influence U's choices are the same as he or she perceived from $\mathcal{H}$, i.e. contextual data that are both perceived by U and relevant for the score. We leave the question of contextual data influencing U's choice but not relevant to the score for future research. Thus, we propose to consider D_{U} as a function that, given a perceived context, returns the choice of an item $i \in I$, hence:

$$D_{\mathsf{U}} : \begin{cases} R_U \to I \\ \widehat{H} \mapsto D_{\mathsf{U}}(\widehat{H}) \end{cases} \tag{5.3}$$

The problem of finding the best possible behaviour $D^{\star}_{\mathsf{U}}$ for U can then be expressed as the following:

$$D^{\star}_{\mathsf{U}} = \underset{D_{\mathsf{U}}}{\text{ArgMax}} \left\{ \sum_{\widehat{H} \in R_{\mathsf{U}}} P(\widehat{H}) \sum_{H \in \mathcal{H}} P(H|\widehat{H}) \sum_{i \in I} P(D_{\mathsf{U}}(\widehat{H}) = i) S(i, H) \right\} \text{ with : } \widehat{H} = \mathcal{M}(H) \tag{5.4}$$

The best behaviour $D^{\star}_{\mathsf{U}}$ for the user is the one that, for each possible perceived context $\widehat{H}$, maximizes the expected score of the chosen action $D_{\mathsf{U}}(\widehat{H})$ in the actual context H, depending on the probability of encountering $\widehat{H}$ and on the probability for the actual context to be H when observing $\widehat{H}$

As one can notice, finding the best behaviour for U depends on both the score function S on the one hand and the mapping function $\mathcal{M}$ and the representation space R_{U} of U on the other hand. Thus in the following, we propose to investigate the question of the best behaviour for U for a given mapping and representation space.

Particularly, we focus on score functions taking into account the context in the form of historical data. Indeed, as presented in 4.5, such scores exist and are even widely used in the nutrition domain. Moreover, this type of score function poses questions that are complementary to those addressed in the contextual recommender system literature. One can especially notice that historical context, which can be referred to as past choices

sequence, is strongly influenced by each item choice. As evoked in 5.1.1, as actions influence the environment, they influence context. This kind of context is particularly interesting in the coaching scenario, as recommendations are aimed to influence users' actions and so may have an impact on future contexts encountered by users.
In other words, we focus on making recommendations for behaviour change in the case of contextual evaluation of behaviour and where performed behaviour has an impact on future contexts.

5.3.2. Case of historical context

Here we consider a score function taking into account the history of U's item choices as a context to evaluate a given choice. In other words, the value associated with a given item $i \in I$ depends on the past choices sequence of the user. In the following, we consider the length of the sequence or the number of considered past choices as a parameter of the score function S. Then the set of all possible contexts is $\widehat{H} = I^m$ hence the notation:

$$S_m : \begin{cases} I \times I^m \to \mathbb{R} \\ (i, H) \mapsto S_m(i, H) \end{cases} \tag{5.5}$$

We propose to model the capacity of the user to represent the context he/she can perceive as a maximum sequence length that he or she can remember. The underlying idea is that a user's memory is limited and that the more recent a choice, the easier it is for a user to remember it. When considering the case of food choice, one may remember his/her last two or three meals, but it seems much less likely that he or she remembers all the meals of the last ten days. This assumption gives us both a user representation space R_{U} and a mapping from $\widehat{H}$ to R_{U}.
Thus, we can denote $R_{\mathsf{U}} = I^\mu$ with $\mu \in \mathbb{N}$ a parameter of U defining the number of past choices U takes into account when making the choice of an item. Hence the user decision function D_{U} is defined as:

$$D_{\mathsf{U}} : \begin{cases} I^\mu \to I \\ \widehat{H} \mapsto D_{\mathsf{U}}(\widehat{H}) \end{cases} \tag{5.6}$$

Then the problem of "incomplete information" discussed in 5.1.3 can be expressed as follows:

- If $\mu \geq m$ then $card(R_{\mathsf{U}}) \geq card(\mathcal{H})$. The user can represent all the contexts needed to compute S_m.
- If $\mu < m$ then $card(R_{\mathsf{U}}) < card(\mathcal{H})$. The user representation space does not allow him/her to represent all the contexts needed to compute S_m. At each step, the user faces the problem of incomplete information and only has partial information about the current context.

As denoted above, historical context is a type of context that is particularly impacted if not defined by the decisions previously taken. In that sense, a particular action impacts the future contexts and so the future actions. In other words, an action at time step t does not only impact the score at time step t but also the future score values.
Given the interdependence of context and action, evaluating the interest of a given action in terms of the score is not straightforward. We propose to model the value Ψ of each item choice made by U in a given context c as an expected return, as defined in [59], hence:

$$\forall\ (i, H) \in I \times \mathcal{H} :\ \Psi(i, H) = S_m(i, H) + \sum_{j \in I} P(D_{\mathsf{U}}(f_i(H)) = j)\Psi(j, f_i(H)) \tag{5.7}$$

with $f_i(H)$ the next context, resulting from choosing item i when in context H. This poses problems in terms of evaluation, as the dependence of $\Psi(i, H)$ over $\Psi(D_{\mathsf{U}}(f_i(H)), f_i(H))$ may result in an infinite return. However, if we make the assumption of a finite return, at each time step $t \in T$, a perfectly taught user, observing the context $\widehat{k}$ should choose item $j^\star$ such as:

$$\forall\ \widehat{H} \in R_{\mathsf{U}} : j^\star = \underset{j \in I}{\text{ArgMax}} \left\{ \sum_{H \in \mathcal{H}} \Psi(j, H) \times P(H|\widehat{H}) \right\}$$

with $P(H|\widehat{H})$ the probability of being actually in context H when observing $\widehat{H}$. As one can notice, one main limitation makes it difficult for the user to actually learn a decision function that ensures him/her to choose $j^\star$ at each time step t. It comes from the term $P(H|\widehat{H})$. Indeed, depending on the representation space of U, this term may be difficult to compute:

- If $\mu \geq m$ then $\widehat{H}$ contains more information than H and given our formalization of context as historical data, we can note that for all contexts $\widehat{H} \in R_{\mathsf{U}}$, it exists a unique context $H \in \mathcal{H}$ such that $P(H|\widehat{H}) = 1$

- Conversely, if $\mu < m$ then $\widehat{H}$ contains less information than H. In other words, U cannot know with certitude from his/her observation $\widehat{H}$ what is the context H that he or she is currently in. As historic is dependant upon the actions taken by U, $P(H|\widehat{H})$ depends both on previously encountered contexts and user decision function. Given the interdependence of contexts and decisions, it appears difficult to compute or estimate $P(H|\widehat{H})$.

Moreover, we have no clue that the assumption of finite return is respected here. Thus, we propose in the following section a solution to this problem that both allows the user to eventually have access to $P(H|\widehat{H})$ and assures that for all couples (i, H) the value $\Psi(i, H)$ is finite.

5.3.3. Proposed Method

5.3.3.1 Notion of deterministic *routine*

The solution we propose is based on the idea of making it possible for U to know with certitude the actual context H he or she is currently facing when observing the perceived context $\widehat{H}$. In other words, we propose a solution that guarantees for a given observed context $\widehat{H}$ that there exists a unique context H such as $P(H|\widehat{H}) = 1$.
The idea we propose is for U to implement a behaviour that follows a given *routine*, that is, a behaviour that repeats over time in a deterministic way. Thus, the idea is for U to associate with each observed context a unique possible decision. Given a considered observed context, the user following a *routine* will always act the same, leading to a unique new context, in which the action will also be deterministic, hence the notion of *routine*: the user follows a path in the space of observed contexts, fully determined by its decision function. We note:

$$\forall\ \widehat{H} \in R_{\mathsf{U}},\ \exists!\ i \in I \text{ such that } P(D_{\mathsf{U}}(\widehat{H}) = i) = 1$$

The power of this approach lies in the fact that when considering historical contexts, the future observed contexts are defined by the previous choices. Then, by making choices in a deterministic way, the set of contexts visited by the user and their appearance probability can be easily computed. Moreover, as the set of perceived contexts R_{U} is finite, following a deterministic *routine* implies that U eventually draws a cycle in R_{U}, where every context $\widehat{H}$ is visited at most once in a cycle full revolution. See Figure 5.2 for an illustration.

Given this, we can notice that the sequence formed by the consecutive choices of a user following a deterministic routine is eventually a periodic sequence. Let us denote (u_n) such a periodic sequence. We have $u_{n+p} = u_n$ with p the period of the sequence, that is, the number of actions taken by the user before going back to the first action forming the periodic sequence. Now if we consider a sequence (v_n) such as $v_n = (u_n, u_{n+1}, ..., u_{n+m})$. It appears that $v_{n+p} = (u_{n+p}, u_{n+1+p}, ..., u_{n+m+p}) = (u_n, u_{n+1}, ..., u_{n+m}) = v_n$. So (v_n) is also a periodic sequence with the same period p. In other words, as it exists a finite size m for the historical context, the sequence formed by the consecutive contexts of size m is eventually periodic, with the same period p as the sequence of observed contexts.
So it appears that following a deterministic routine lets U eventually draw a cycle in $\mathcal{H}$, with the same period as the cycle drawn by U in R_{U}. In other words, such a deterministic decision function generates a behaviour that eventually loops, hence the notion of *cycle* in the context set. This idea of looping in the set of observed contexts has two major consequences. First, the context perceived by U, namely $\widehat{H}$, gives an indirect but perfect knowledge of the corresponding real context H: once the user is in the cycle, it exists a unique context H associated with the observed context $\widehat{H}$. Second, one can easily compute a mean score value associated with a cyclic behaviour when repeated indefinitely, hence the solution to the problem of the infinite return value. By considering the score gains on a full cycle revolution and the length of that revolution, one can compute the mean gain for the user to repeat a given cyclic behaviour indefinitely.
Le

5.3.3.2 Finding the best cyclic behavior

Given the existence in the observed context set of cycles that are each associated with a deterministic routine, the objective is to find the cyclic behaviour for the user that maximizes the mean score value of the cycle. Given that such a cycle can be fully defined by the sequence of observed contexts and the associated action, one way of solving this problem is to draw a directed graph G in which the vertices set is the set of all possible observed context R_{U}, namely I^{μ}, and where an edge goes from vertex $, \widehat{H_1}$ to vertex $\widehat{H_2}$ if and only if $\exists i \in I$ such that $\widehat{H_2} = f_i(H_1)$, that is it exist an item that by being chosen in context H_1 leads to context H_2. Then the set of all elementary circuits in G is the set of all the possible cycles resulting from the routines that can be implemented by a user U given I, m, and μ.
Figure 5.2 presents an example of such a graph for $I = \{a, b\}$ and $\mu = 2$. In this case, it

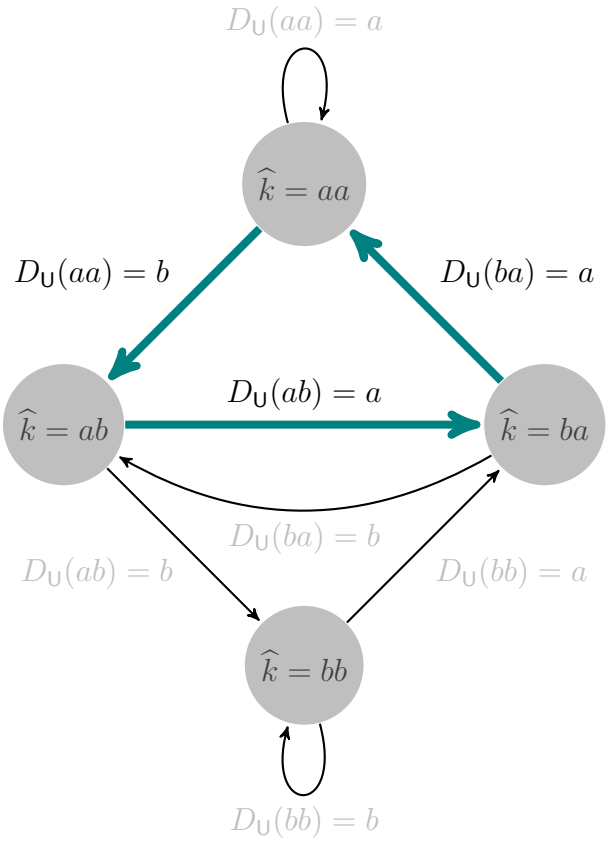

Figure 5.2: Example of consumption routine, resulting in a cyclic behaviour. Here $I = \{a, b\}$ and $\mu = 2$. Thus the representation space of the user is composed of four possible perceived contexts, and $R_{\mathsf{U}} = \{aa, ab, ba, bb\}$.

exists six different cycles c that can result from following a deterministic routine for the user:

$$\begin{cases} c_1 = (\widehat{H} = \{aa\},\ D_{\mathsf{U}}(\widehat{H}) = a) \\ c_2 = (\widehat{H} = \{bb\},\ D_{\mathsf{U}}(\widehat{H}) = b) \\ c_3 = (\widehat{H} = \{ab\},\ D_{\mathsf{U}}(\widehat{H}) = b) \to (\widehat{H} = \{bb\},\ D_{\mathsf{U}}(\widehat{H}) = a) \to (\widehat{H} = \{ba\},\ D_{\mathsf{U}}(\widehat{H}) = b) \\ c_4 = (\widehat{H} = \{ab\},\ D_{\mathsf{U}}(\widehat{H}) = a) \to (\widehat{H} = \{ba\},\ D_{\mathsf{U}}(\widehat{H}) = a) \to (\widehat{H} = \{aa\},\ D_{\mathsf{U}}(\widehat{H}) = b) \\ c_5 = (\widehat{H} = \{ab\},\ D_{\mathsf{U}}(\widehat{H}) = b) \to (\widehat{H} = \{bb\},\ D_{\mathsf{U}}(\widehat{H}) = a) \to \\ (\widehat{H} = \{ba\},\ D_{\mathsf{U}}(\widehat{H}) = a) \to (\widehat{H} = \{aa\},\ D_{\mathsf{U}}(\widehat{H}) = b) \\ c_6 = (\widehat{H} = \{ab\},\ D_{\mathsf{U}}(\widehat{H}) = a) \to (\widehat{H} = \{ba\},\ D_{\mathsf{U}}(\widehat{H}) = b) \end{cases}$$

For each of those, when considering a score function S_m with parameter m, defined from

$I \times I^m$ into $\mathbb{R}$, one can compute the associated cycles in $\mathcal{H} = I \times I^m$. Then for each cycle, we can compute a *cycle mean value* V_c defined as:

$$V_c = \frac{1}{card(c)} \sum_{(i,H) \in c} S_m(i, H) \tag{5.8}$$

Then all the decision functions D_{U} that eventually lead to the cycle $c^\star$ that maximizes V_c are guaranteed to be the best optimal deterministic decision function. The question that arises then is how to efficiently find the cycle(s) maximizing V_c.

Brute force method

The first considered method in order to find $c^\star$ is to enumerate all cycles in G and to compute their associated value V_c.

There exist in the literature various papers containing algorithms enumerating all cycles in a directed graph [194] such as Johnson's [195] or Szwarcfiter and Lauer's [196]. According to [194], Johnson's algorithm is the fastest for this task with a time complexity bounded by $O((n+e)(c+1))$ where n is the number of nodes in the graph, e the number of edges and c the number of cycles.

In our case, we can easily compute the number n and e of edges from R_{U} as, by definition, the nodes of the historical graph G are the elements of R_{U}.

In the case of historical context, one can note that we have $R_{\mathsf{U}} = I^\mu$ so $n = card(R_{\mathsf{U}}) = card(I)^\mu$. Moreover, as D_{U} is defined from R_{U} into I, each node of the graph is associated with $card(I)$ edges, pointing towards all the possible contexts resulting from choosing an item$i \in I$ in context $\widehat{H} \in R_{\mathsf{U}}$. Hence $e = n \times card(I) = card(I)^{\mu+1}$. Thus the time complexity to find all cycles in G is bounded by $O(card(I)^\mu(card(I)+1)(c+1))$. In other words, the time complexity growth is *at least* quadratic in $card(I)$ and exponential in μ. In addition, if we experimentally investigate c the number of cycles in the graph depending on μ and $card(I)$, we can observe that it grows extremely fast, as denoted in table 5.1.

Thus it appears that computing all the possible simple cycles is only possible for a very limited number of cases. As such brute force method appears to be extremely limited when considering actual coaching situations, where the number of items may be large, as well as the user memory.

Proposed practical heuristic

$card(I)$ \ μ	1	2	3	4
2	3	6	19	179
3	8	148	3382522	NC
4	24	120538	NC	NC

Table 5.1: Table presenting c the number of elementary circuits existing in graph G depending on $card(I)$ and μ. NC stands for "Not computed": the computation time being too long.

Instead of pre-computing all possible cyclic behaviours, we propose a method to take advantage of being in an interaction with the user to reduce the number of explored cycles. The idea is to observe the behaviour of the user and to compute cycles based on the contexts actually visited by U. Indeed, most of the time, a user will only live in a limited subspace of the context representation space R_{U}. Moreover, knowing the contexts visited by the user and the substitutability values of items for this user, a coach can compute the contexts that are possibly reachable for this user. Thus we propose a recommendation algorithm based on this idea: the coaching interaction lies in two phases. In the first phase, the coach observes the behaviour of the user and builds a context graph with the observed contexts and transitions. At the end of the first phase, C enrich the graph by considering the substitutability between items: for each vertex $\widehat{H}$ of the graph, consider its successor vertex (if it exists) formed by choosing item i in context $\widehat{H}$. If there exist items that can be substituted to i, then the contexts formed by choosing it instead of i in context $\widehat{H}$ can be added to the contextual graph. The size of the formed graph, however, stays limited and thus lets us compute the set of cycles. Notice that these two phases can be alternated along the interaction so as to discover a greater number of nodes and, thus, cycles. Once the best cycle in this sub-graph is found, the coach can follow a recommendation strategy, as we will see in 5.4, that eventually leads the user towards that cyclic behaviour. Algorithm 2 presents the interaction procedure described above.

5.4. Best recommendation strategy towards target behaviour

The second question we focus on in this section is the one of making recommendations to a user that promote a change from his/her behaviour towards a given target when consid-

begin
  $H = initial\_state$
  C initializes G as an empty graph
  while *coaching in play* **do**
    1st phase: Observation
    while *1st phase in play* **do**
      U proposes item $i = D_{\mathsf{U}}(\widehat{H})$
      C does not make any suggestion
      C adds $\widehat{H}$ to G
    end
    $H = f_{i \to i}(H)$
    2nd phase: Graph augmentation
    for *node in G and i in Items* **do**
      if $substitutability(last\_item(node), i) > threshold$ **then**
        for *p in G.predecessors(node)* **do**
          C adds edge(p, replace(node,i)) to G
        end
      end
    end
    C computes Cycles(G) with Jonhson algorithm [195]
    ; $c^\star = \text{ArgMax}_{c \in Cycles(G)}\, mean\_value(c)$
    3rd phase: Cycle recommendation
    while *3rd phase in play* **do**
      U proposes item $i = D_{\mathsf{U}}(\widehat{H})$
      C suggests substitution $i \to j = cycle\_rec(i, H)$
      U accepts the substitution $i \to j$ with probability $m_{i,j}$.
      H updates
      C updates $\widehat{m_{i,j}}$
    end
  end
end

Algorithm 2: Cycle identification algorithm based on the contexts actually encountered by the user, and reachable given the substitutability matrix of U. The cycle recommendation algorithm *cycle_rec* is the subject of section 5.4

ering the user choice contextual. We investigate the question of finding a recommendation strategy in this setting.

5.4.1. User model

As stated in the chapter introduction, contextual factors influence human decision-making. Thus, it appears essential to consider the context in the coaching framework. Moreover, the user model proposed in 3 needs to be updated in order to take into account the impact of context on user decisions. Indeed in Chapter 3, we modelled the habits of the user by a probability vector, leading to a decision function independent from any notion of context. Thus we propose an updated version of the user model that takes into account the context of the user decision function. As the previously presented one, it consists of three components:

- As in Section 5.3.2, we consider the consumption history as the contextual information taken into account by the user. We propose to use the same concept of *choice vector* but incorporating the idea of context. Thus we propose a model where for each possible context the user can encounter, he or she has a corresponding choice vector, determining the probability of each item choice. Then the decision function of the user at each time step t is modelled as a random draw in I following the corresponding choice vector $\Pi_{t,\widehat{k}}$. We note :

$$\forall t \in T,\ \forall \widehat{k} \in R_{\mathsf{U}},\ \text{we have } \Pi_{t,\widehat{k}} : \begin{cases} I \to [0,1] \\ i \mapsto \pi_{t,k}(i) \end{cases}$$

- We make the hypothesis, as previously, that the acceptability for the user of a substitution from an item $i \in I$ to an item $j \in I$ does not depend on time. Moreover, we make the additional hypothesis that acceptability neither depends on context. Hence the modelization of the user's acceptability of substitutions through a matrix $M : I \times I$ associating to each couple of items $(i,j) \in I^2$ a coefficient $m(i,j) \in [0,1]$ representing the probability that user U accepts the substitution $i \to j$ from i to j.

- When considering user behaviour change, we make the assumption that, as user choice, it fully depends on the context. Thus we consider that the acceptance of a change in a given context only impact the future choices of U in this particular

context. We note:

$$\begin{cases} \pi_{\widehat{k},t+1}(i) = (1-\lambda)\pi_{\widehat{k},t}(i) \\ \pi_{\widehat{k},t+1}(j) = \pi_{\widehat{k},t}(j) + \lambda\pi_{\widehat{k},t}(i) \\ \pi_{\widehat{k},t+1}(k) = \pi_{\widehat{k},t}(k)\ ,\ \forall k \in I \setminus \{i,j\} \end{cases} \quad \text{If } i \to j \text{ has been accepted in context } \widehat{k} \in R_{\mathsf{U}}$$

with $\lambda \in [0,1]$. As in 3.1.3, $\Pi_{\widehat{k},t+1}$ is guaranteed to be a probability distribution if $\Pi_{\widehat{k},t}$ is so.

Furthermore, we consider the same model for the coach as the one described in 3.1.4. We still model the interaction as a two-player iterated game.

5.4.2. Step by step context induction

The recommendation problem faced by C is then to make recommendations that will lead U from his/her current behaviour towards the targeted behaviour. It is worth noting that this recommendation problem differs from the one addressed in Chapter 3. Indeed, in Chapter 3, we proposed an optimal choice criterion based on long-term score gain. The recommendation strategies we tested were derived from this criterion and tried to maximize an approximation of that criterion rather than focusing on a particular behaviour. On the contrary, here, the coach knows exactly the target behaviour and should make recommendations that encourage the user to change his/her behaviour so that it eventually becomes the targeted behaviour.

In this setting, what should guide the recommendation is no longer a value such as expected gain or Q-value but a notion of distance between the current user behaviour and the targeted one. Thus, if we consider a distance metric d, we can note $d(\Pi_t, \Pi^\star)$ the distance between the user behaviour at step t, noted Π_t, and the target behaviour $\Pi^\star$. Then a valid recommendation strategy for the coaching system should ensure that, as the interaction last, this distance converges to zero. Moreover, an optimal recommendation strategy should achieve this convergence in a minimal time. Solving this problem theoretically appears unlikely to be possible, except for the simplest cases. Consequently, we instead focus on finding a valid strategy applicable to our user model.

We propose a heuristic recommendation strategy based on the least resistance path. The

underlying idea is to consider two possible cases for the recommendation, depending on the observed user behaviour. Consider a given cyclic behaviour that is the target behaviour towards which the user should strive. This behaviour can be represented as a set of encountered contexts, each associated with the action that must be taken in this context. Then we consider two distinct cases:

- U is in a context that forms part of the set of encountered contexts associated with the target behaviour. In this case, the recommendation to make is straightforward: the coach should recommend substituting the user's initial choice with the item associated with that context in the targeted behaviour.

- U is in a context that does not belong to the consumption cycle forming the target behaviour. In that case, we propose that the coach takes advantage of the impact of its recommendation (if accepted) on future contexts. Indeed, the coach, by having its recommendation accepted, can lead the user closer to a context that belongs to the target behaviour. Let us consider the example presented in 5.3.3.2: we have shown that in the case where $I = \{a, b\}$ and $\mu = 2$, there exist six possible cyclic behaviours that can be implemented by a user U. We consider as the target behaviour the cycle: $(\widehat{H} = \{ab\},\ D_{\mathsf{U}}(\widehat{H}) = a) \rightarrow (\widehat{H} = \{ba\},\ D_{\mathsf{U}}(\widehat{H}) = b)$. Then if U is in observed context $\widehat{H} = \{bb\}$ and is willing to choose item b, the coach should recommend substituting item b by item a: if accepted this change will lead U to the context $\widehat{H} = \{ab\}$, which forms part of the targeted cyclic behaviour, and so for which C can recommend the corresponding action in the target behaviour.

Thus we propose an algorithm based on these two cases, whose objective is to build step by step the contexts of interest, that is, the contexts forming part of the target behaviour. Once the user is in a context belonging to the target behaviour, the coach recommends the action corresponding to that context in the target behaviour. In other words, for each possible context $\widehat{H} \in R_{\mathsf{U}}$, there is a target item j that should be recommended by C to lead U to the targeted behaviour, regardless of U's proposal i in that context. However, U may not be ready to substitute i by j. The question that arises then is: how should the coach take into account the substitutability of items in its recommendations?

5.4.3. Taking in account substitutability values

The problem of recommendation faced by the coach is the following: given a proposed item i by a user U and a target item j that the coach wants U to choose, what substitution should recommend the coach? We propose to base the recommendation on the *least-resistance path*. The idea is to make acceptable recommendations that eventually lead U from i to j. For example, consider the context of a nutritional coach interacting with a user to recommend food items. The user proposes to eat french fries, while the coach has identified carrots as the target food item in that context. However, substituting french fries with carrots is not acceptable for the user. A solution for the coach is to consider an indirect substitution, for example recommending substituting french fries with steamed potatoes, and then once the user proposes steamed potatoes, recommending substituting with carrots.

Such indirect paths can be obtained by drawing a substitutability graph from the substitutability matrix of the user: consider each item $i \in I$ as a vertex of the substitutability graph. A directed edge from vertex i to vertex j exists if and only if i can be substituted with j, that is, $m(i,j) > 0$. Then the set of all possible indirect substitution paths from an item i to an item j is the set of all paths in the graph from vertex i to vertex j. Moreover, let us consider a weight w associated with each edge of the graph from a vertex i to a vertex j, representing the difficulty for the user to substitute i by j. For example $w_{i,j} = \frac{1}{m(i,j)}$. Then the weighted shortest path from i to j represents the *least-resistance path* for the user from item i to item j. So we propose that the coach make recommendations that follow this *least-resistance path* when a direct recommendation is impossible. Hence the algorithm 3.

5.5. Discussion

In this chapter, we proposed to consider the question of coaching recommendations in a contextual case. In particular, we focused on the case where the score function of the coach, which allows measuring the quality of a user's behaviour, is contextual. That is, depending on the context, the interest in a given item may change, and a highly valuable item in a given context may be much less interesting in a different context.

Considering this setting, we proposed to study two questions arising from it: first, we investigated how to maximize such a score from the user side. We proposed a method

begin
  $L = \mu$;
  Possible_Recomendation_List initialized as an empty list **while** *recommendation not found* **do**
    for $(context, action)$ *in target_behavior* **do**
      if *last_L_elements(context) == last_L_elements*$(\widehat{H})$ **then**
        append action to Possible_Recomendation_List
      end
    end
    if $Possible\_Reco \neq \emptyset$ **then**
      if $Max(m(\hat{i}, r)$ *for r in Possible_Recomendation_List)* $>$ *threshold* **then**
        $recommendation \leftarrow \text{ArgMax}_{\hat{m}(i,r)}\, Possible\_Recomendation\_List$
      else
        $recommendation \leftarrow \text{ArgMin}_{r \in R}\, LeastResistancePath(i, r)$
      end
    else
      $L \leftarrow L - 1$
    end
  end
end

Algorithm 3: The cycle recommendation algorithm. The principle of the recommendation is to build step by step, by having the recommendation accepted, the contexts that form part of the targeted behaviour.

based on repeating routines that allows a user to learn behaviour in his/her representation space that maximizes the score function regarding his/her capacities, even though he or she is not able to consider directly the features needed to compute the score.

Second, we propose a heuristic algorithm for a coaching system to make recommendations that lead a user toward a specific target behaviour. This algorithm is based on step-by-step context induction to lead the user to a situation where he or she can learn the target behaviour.

However, we left aside the question for the coach to learn the representation space of the user. Indeed, if the coach is able to compute the best deterministic routine and to make recommendations that eventually lead a user towards this routine, this necessitates knowing the representation space of the user. Hence the following question: How a coaching system could learn from its interaction with a given user the latter's representation space? This is a tough problem, as the representation space of the user cannot be directly observed by the coach. It poses two main questions: first, the question of the user representation space modelling, and second the question of the data contained in the user feedback.

As we proposed to focus on the historical context, we considered the representation space of the user through a given "memory" parameter μ determining the number of passed choices the user takes into account when making a choice. Several points are worth discussing regarding this setting. First, we considered the memory evenly distributed. That is, a user remembers each passed choice with the same intensity, i.e. for the same time. However, one can argue that some choices are more "significant" than others and thus may be remembered for a longer time. As an example, consider the task of choosing meals. The last weekend's feast may be more significant, and its impact on future choices may last longer than the classic sandwich of yesterday's lunch.

Similarly, when playing tennis, the surprising under-arm serve that led to victory in the last set may have a longer impact on the decision than the classic forehand shot of the last point. Even if this assumption makes sense, having access to these different levels of memory appears to be very difficult. Moreover, computing the relative importance of the corresponding significance levels appears even more problematic. However, it is noticeable that conceptually, considering different levels of memory does neither refute the proposed model nor the idea of consumption cycles. Indeed, we can consider the general memory of the user as a combination of these different levels. As such, it could be possible to identify different cycles of consumption for each level of significance and to propose a

solution combining these different cycles. For example, the authors of [197] proposed a recommendation framework taking into account two levels of memory: short-term and long-term memory. Moreover, considering different levels of memory would potentially make it possible to capture other dynamics, for example, seasonality in food choice.

Second, we assumed that the contextual data that influence the user choice are the same as he or she perceived from the set of real contexts influencing the score function. This assumption is key as it allows the user to learn a contextual behaviour that depends on an observed fraction of the real context. One can argue that the set of contexts influencing the user choice may be fully disjointed from the set of contexts having an impact on the score function, hence independence between the context observed by the user at time t and the contextual features needed to compute the score. However, it is noticeable that this case can be seen as a classical CARS problem. Indeed, in that case, the set of contexts can be seen as an additional set of features, as in the majority of CARS approaches. But if this set is really independent of the set of real contexts, the context observed by the user has no impact on the score. If these two sets are not really independent, then the relations that exist between the observed contexts and the real contexts relevant to the score can be expressed as a user mapping function from the set $\mathcal{H}$ to a user representation space $R_{\mathcal{U}}$. Thus the objective of the coach is still to make recommendations to the user such as the latter "learns" a representable behaviour that maximizes the score function.

However, this question also highlights the problem for the coach of learning the representation space of the user from the available feedback. An eventual research direction for this problem could be found in the literature about "concept drift" detection. Indeed, it exists an extensive body of literature about methods to detect changes in the underlying distribution of streaming data [198]. Such methods could potentially be used to detect changes in the behaviour of a user over time. Studying how the behaviour evolves could give hints about the representation space the user lives in.

Another direction for representation space learning, which is complementary to a concept drift detection approach, is to enrich the interaction between the user and the coach and, in particular, to allow the coach to have access to more data about the user. If the exact form of such an enriched interaction is not necessarily easy to define, it appears important for the coach to gather more data than the only proposal of the user and acceptance or refusal of the recommendation. In particular, if we consider a parallel with a human-human coaching interaction, it could be helpful for the coach to ask the user information about how he or she learns and what he or she feels able to remember or not.

To conclude, we proposed in this chapter a formalization of the problem of coaching under contextual evaluation. We also proposed heuristics to identify a target behaviour and make recommendations towards it, given a representation space of the user. In future works, we are willing to test our proposed methods in a modelled environment with a real contextual score function, such as, for example, the INCA2 dietary quality index. Moreover, we are interested in developing methods to infer from the coaching interaction the representation space of the user and to test the contextual coaching in a real-world setting.

6. Conclusion

Recommender systems have gained in popularity over the past decades and have become pervasive in most of our everyday life. They accompany an increasing part of our choices online. Consequently, they affect how we make choices and, thus, our behaviour and habits. This impact of recommender systems on our behaviour raises major concerns about democracy and content diversity. This work describes a recommendation framework we named *coaching*, the aim of which is to take advantage of the impact of recommendations on users' behaviour to accompany the latter in a behaviour change process.
This chapter first summarizes the work conducted in this PhD thesis and the corresponding main contributions. Second, we present the future research directions and perspectives that stem from the proposed formalism and the obtained results.

6.1. Summary and contributions

We first presented a thorough review of the literature on problems related to coaching. Indeed, to the best of our knowledge, specific literature on recommender systems for behaviour change has yet to exist. Thus the literature is conspicuously fragmented, and research communities from diverse domains investigated parts of the problem. We focused on multi-stakeholder recommender systems, teacher-student learning framework and persuasive technologies. We showed that each of these three domains informs us on specific stakes of recommendation for behaviour change. However, we did not find an acceptable framework to model our problem in the literature, as each specific domain solution ignores some crucial part of our question. Consequently, we concluded on the necessity of developing a framework dedicated to recommendations for behaviour change.

Second, we investigated the research question RQ1 presented in Chapter 1: *How to design a recommendation framework that promotes long-term behaviour change of its users ?* We proposed to model the recommendation problem as an iterated two-player game, where a coach agent (the recommender) proposes a recommendation in reaction to the user's

announced choice. By suggesting *substitutions* the coach aims to have the user modify his or her consumption habits. This model encompasses three crucial points: user involvement, personalisation of the recommendation and user behaviour evolution. Moreover, we introduced the notion of *trajectory* in the space of user habits. We investigated the related recommendation problem and showed its specificity compared to classic recommendation approaches. We formally studied the problem and deduced an optimal recommendation criterion from which we derived approached recommendation strategies. The resulting strategies were tested experimentally on a simulated healthy-food recommendation task. The corresponding results highlight the importance of personalisation of the recommendation and the interest in non-myopic strategies. In addition, these results validate the developed framework's efficiency in formulating recommendations for behaviour change.

Thirdly we focused on our second research question RQ2: *How should be designed an automated coaching system to make acceptable recommendation in the food domain ?* We identified two key concerns when considering the real-world applicability of coaching for food recommendation, namely the cold-start problem and the design of the interaction in a coaching scenario.

The former refers to the problem of finding meaningful recommendations when data on user preference is unavailable. We designed an experiment to test the capacity of a previously proposed method to extract meaningful substitutability relations between food items from consumption data when interacting with real users. Results show that even if the method performs significantly worse than a domain expert, it can extract some meaningful relationships from data.

The latter interrogates how an automated coach should interact with a user to maximize the acceptability of its recommendation and, therefore, to maximize its impact on the user. We designed a second experiment with real users and tested three modalities of recommendation. The results indicate that the user's involvement in the recommendation process is a key determinant of recommendation acceptance.

Finally, we investigated the research question RQ3: *How could an automated coaching system incorporate context and temporal dynamics in the evaluation of its recommendations ?* We proposed to divide this research question into three sub-questions, relative to finding the best user behaviour, making recommendations towards a given target behaviour and inferring the user learning capacity.

Regarding the sub-question of finding the best user behaviour, we formalized the general problem of finding the best representable contextual behaviour for a user and studied the particular case of historical context. We proposed a method to compute from the user characteristics a representable behaviour that maximizes a given contextual score function. Moreover, we showed that the complexity of finding the exact solution to this problem makes it impossible to compute in most real-world cases. Thus we proposed a heuristic algorithm to approach the solution.
We then focused on the second sub-question of making recommendations to lead a user towards a given target behaviour. We proposed a heuristic recommendation algorithm based on step-by-step context induction. However, the question of the actual best recommendation strategy, in this case, is still to be explored, and the proposed algorithm can be seen as a preliminary step in the investigation of this question.
We left aside in this work the third sub-question. Nevertheless, it appears crucial in the setting of contextual coaching and calls for in-depth investigation.

6.2. Perspectives

The work presented in this PhD thesis investigated the question of making recommendations with an objective of long-term behaviour change and proposed a corresponding framework we called the *coaching framework*. We consider this work as a first step in the investigation of *automated coaching systems*, which we strongly believe to correspond to a large spectrum of recommendation problems. Indeed, our contributions do not cover all the challenges and questions raised by automated coaching. By presenting the following perspectives, our objective is to highlight remaining meaningful challenges, propose potential solutions and point out possible future research directions.

6.2.1. Explainability of the recommendation

As shown in [199], human students learn more efficiently when the teacher provides explanations. Regarding this, we can assume that improving the explainability of recommendations may positively influence the behaviour change of the coached users. Hence the interest in explainable recommendations in coaching. For example, the authors of [200] present, in the possible aims for explanations in recommender systems, the stakes of trustworthiness, effectiveness and persuasiveness. These appear to be particularly rel-

evant in the case of recommendations for sustainable behaviour change. Explainability of recommendations is a problem that has been widely studied in the classical recommender systems literature [201].

It is noticeable that, in coaching, the explanations should encompass both the acceptability aspect and the relation with the user behaviour change objective. On the one hand, the acceptability aspect of explanations can be assimilated into the question of the explainability of recommendations in traditional recommender systems. Indeed, most of the state-of-the-art methods are only based on user preferences, and methods have been developed to provide explanations on why an item should match user preferences. In particular, content-based approaches allow explaining recommendations based on item features, which appears to be promising for application cases such as food recommendations.

On the other hand, the proximity to the user objective is represented, in coaching, by the score function. So a recommendation should also be explained through the lens of score gains. This raises the question of the interpretability of the score function. Indeed, the actual score gain may not be sufficient to explain the recommendation clearly. Additional commentaries could be beneficial to favour the user's understanding. This would necessitate working with domain experts, who might interpret the score variations. Moreover, we have seen in the work presented in Chapter 3 that non-myopic strategies are the most valuable in certain cases. Another aspect of the recommendation's explainability could be to present to the user the expected steps that have to be reached. This can be linked with the problem of path recommendation, which has been investigated in the literature.

6.2.2. Learning path recommendation

A line of work in the literature, particularly in intelligent tutoring systems literature, is interested in path recommendation and, more precisely, learning path recommendation. Works on this topic consider learning as a sequential task from a given starting point to a predefined objective [202]. Then the recommendation consists in recommending the user to follow a path in the space of items (traditionally courses or exercises) that eventually leads him or her towards the predefined objective. This approach appears to be relevant to the idea of coaching, particularly in the case of contextual coaching. Indeed, as we have seen in Chapter 5, the problem of contextual coaching implicates the

question of recommendation towards a specific behaviour. Moreover, if we proposed a method to compute a representable behaviour for the user, we did not explicitly consider the difficulty of reaching this behaviour regarding the user's current state.

By building a graph representation of the possible items to be recommended, works on learning path recommendations allow finding personalized recommendation sequences adapted to the pursued goal. For example, the work presented in [203] proposes to learn the best recommendation paths for users based on their initial results to a preliminary test. This could be applied to contextual coaching, clustering users based on their initial habits and then finding the best path towards the pre-computed best representable behaviour. Nevertheless, most existing works consider a clear hierarchy of items, some being prerequisites for others. This is due to the very nature of considered items, i.e. learning resources. Such a hierarchy is not necessarily easy to determine in contexts in which coaching could be applied, such as nutrition. Moreover, designing a learning path requires being able to observe how a user learns.

6.2.3. Learning how the user learns

As we denoted above, learning for the coach how the user learns, although little explored in this PhD thesis, is an important stake for long-term behaviour change recommendation. Indeed, whether in the item-based case presented in chapter 3 or the contextual case presented in Chapter 5, the learning capacity of the user influences the best recommendation strategy. The question of estimating both λ, the learning rate of the user, and μ, the user memory determining his/her representation space, is challenging. Indeed, even the user has no direct access to this information.

One possible solution for the coach is then to estimate, from its interactions with the user, the latter's learning parameters. A possible research direction is to consider the methods developed in the literature on concept drift detection. As denoted in [198], "Concept drift describes unforeseeable changes in the underlying distribution of streaming data over time". In the particular case of coaching, the evolution of users' behaviour over time and changes in their habits can be considered a form of concept drift. By tackling it, a coaching system may be able to estimate the impact of its recommendations on future users' behaviours and so to estimate how users learn from their interaction with the coach.

6.2.4. Real-world food recommendation

Alongside the theoretical investigations of coaching systems, we also plan to test coaching in real-world settings. We are particularly interested in applying coaching to the problem of healthy food recommendation. Nevertheless, real-world recommendation poses many problems. One of the main questions is food item accessibility. Whether because of immediate inaccessibility or because of the high cost of the recommendation, the user may not have access to the recommended item and so may be unable to follow the recommendation even though it would have been acceptable for him/her. In particular, the cost of the recommended items is an important point, as recommended foods in the healthy recommendation are usually expensive [204, 205].
An experiment design that circumvents these limitations is to propose a coaching system for food choice in university restaurants. Indeed, in this setting, the set of all possible choices is easily accessible for the coach. Moreover, in the traditional operation of university restaurants, food items for similar roles (starters, main courses or desserts) have the same price. Bypassing the accessibility question could lead to a long-run experiment, making it possible to measure long-term behaviour changes of the users.

Overall this thesis work presented a new framework for recommendation focused on long-term behaviour change. By being at the crossroads of several communities in the literature, the question of recommendation for behaviour change appears to pose many questions. This PhD work is a first step in the investigation of the problem. We addressed some key questions, but many are still to be investigated. We strongly believe that the domain is promising, but there is still a lot to do, both from the theoretical point of view and from the experimental validation. Even if we focused in this thesis on applications in the healthy food recommendation domain, the developed framework is, in my opinion, general enough to encompass various recommendation problems.

Bibliography

[1] S. Wojcicki, "YouTube at 15: My personal journey and the road ahead." https://blog.youtube/news-and-events/youtube-at-15-my-personal-journey/. Accessed: 2023-01-02.

[2] A. Buck, "57 AMAZON STATISTICS TO KNOW IN 2023." https://landingcube.com/amazon-statistics/. Accessed: 2023-01-02.

[3] C. C. Aggarwal *et al.*, *Recommender systems*, vol. 1. Springer, 2016.

[4] T. T. Nguyen, P.-M. Hui, F. M. Harper, L. Terveen, and J. A. Konstan, "Exploring the filter bubble: the effect of using recommender systems on content diversity," in *Proceedings of the 23rd international conference on World wide web*, pp. 677–686, 2014.

[5] N. Helberger, "On the democratic role of news recommenders," *Digital Journalism*, vol. 7, no. 8, pp. 993–1012, 2019.

[6] H. Abdollahpouri, R. Burke, and B. Mobasher, "Managing popularity bias in recommender systems with personalized re-ranking," in *The thirty-second international flairs conference*, 2019.

[7] Y. Liang and M. C. Willemsen, "Exploring the longitudinal effects of nudging on users' music genre exploration behavior and listening preferences," in *Proceedings of the 16th ACM Conference on Recommender Systems*, pp. 3–13, 2022.

[8] S. Hercberg, S. Chat-Yung, and M. Chauliac, "The french national nutrition and health program: 2001–2006–2010," *International Journal of Public Health*, vol. 53, no. 2, pp. 68–77, 2008.

[9] I. Keller and T. Lang, "Food-based dietary guidelines and implementation: lessons from four countries–chile, germany, new zealand and south africa," *Public health nutrition*, vol. 11, no. 8, pp. 867–874, 2008.

[10] R. Z. Wilson, *Neuroscience for counsellors: Practical applications for counsellors, therapists and mental health practitioners*. Jessica Kingsley Publishers, 2014.

[11] A. Sharma, J. M. Hofman, and D. J. Watts, "Estimating the causal impact of recommendation systems from observational data," in *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pp. 453–470, 2015.

[12] Y. Himeur, A. Alsalemi, A. Al-Kababji, F. Bensaali, A. Amira, C. Sardianos, G. Dimitrakopoulos, and I. Varlamis, "A survey of recommender systems for energy efficiency in buildings: Principles, challenges and prospects," *Information Fusion*, vol. 72, pp. 1–21, 2021.

[13] P. Resnick and H. R. Varian, "Recommender systems," *Communications of the ACM*, vol. 40, no. 3, pp. 56–58, 1997.

[14] F. Ricci, L. Rokach, and B. Shapira, "Recommender systems: introduction and challenges," in *Recommender systems handbook*, pp. 1–34, Springer, 2015.

[15] J. Bennett, S. Lanning, *et al.*, "The netflix prize," in *Proceedings of KDD cup and workshop*, vol. 2007, p. 35, Citeseer, 2007.

[16] Y. Song, S. Dixon, and M. Pearce, "A survey of music recommendation systems and future perspectives," in *9th international symposium on computer music modeling and retrieval*, vol. 4, pp. 395–410, Citeseer, 2012.

[17] S. Hammer, J. Kim, and E. André, "Med-styler: Metabo diabetes-lifestyle recommender," in *Proceedings of the fourth ACM conference on Recommender systems*, pp. 285–288, 2010.

[18] J. Borràs, A. Moreno, and A. Valls, "Intelligent tourism recommender systems: A survey," *Expert systems with applications*, vol. 41, no. 16, pp. 7370–7389, 2014.

[19] I. Andjelkovic, D. Parra, and J. O'Donovan, "Moodplay: interactive music recommendation based on artists' mood similarity," *International Journal of Human-Computer Studies*, vol. 121, pp. 142–159, 2019.

[20] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, vol. 35, no. 12, pp. 61–70, 1992.

[21] L. N. Tondji, "Web recommender system for job seeking and recruiting," *Partial Fulfillment of a Masters II at AIMS*, 2018.

[22] G. Shani and A. Gunawardana, "Evaluating recommendation systems," in *Recommender systems handbook*, pp. 257–297, Springer, 2011.

[23] R. Kohavi, R. Longbotham, D. Sommerfield, and R. M. Henne, "Controlled experiments on the web: survey and practical guide," *Data mining and knowledge discovery*, vol. 18, no. 1, pp. 140–181, 2009.

[24] S. Vargas and P. Castells, "Rank and relevance in novelty and diversity metrics for recommender systems," in *Proceedings of the fifth ACM conference on Recommender systems*, pp. 109–116, 2011.

[25] K. Bradley and B. Smyth, "Improving recommendation diversity," in *Proceedings of the twelfth Irish conference on artificial intelligence and cognitive science, Maynooth, Ireland*, vol. 85, pp. 141–152, Citeseer, 2001.

[26] S. Castagnos, A. Brun, and A. Boyer, "When diversity is needed... but not expected!," in *International Conference on Advances in Information Mining and Management*, pp. 44–50, IARIA XPS Press, 2013.

[27] R. Burke, G. Adomavicius, I. Guy, J. Krasnodebski, L. Pizzato, Y. Zhang, and H. Abdollahpouri, "Vams 2017: Workshop on value-aware and multistakeholder recommendation," in *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pp. 378–379, 2017.

[28] R. D. Burke, H. Abdollahpouri, B. Mobasher, and T. Gupta, "Towards multistakeholder utility evaluation of recommender systems.," *UMAP (Extended Proceedings)*, vol. 750, 2016.

[29] H. Abdollahpouri, G. Adomavicius, R. Burke, I. Guy, D. Jannach, T. Kamishima, J. Krasnodebski, and L. Pizzato, "Multistakeholder recommendation: Survey and research directions," *User Modeling and User-Adapted Interaction*, vol. 30, no. 1, pp. 127–158, 2020.

[30] M. Hammar, R. Karlsson, and B. J. Nilsson, "Using maximum coverage to optimize recommendation systems in e-commerce," in *Proceedings of the 7th ACM conference on Recommender systems*, pp. 265–272, 2013.

[31] W. Lu, S. Chen, K. Li, and L. V. Lakshmanan, "Show me the money: Dynamic recommendations for revenue maximization," *arXiv preprint arXiv:1409.0080*, 2014.

[32] W. Liu, J. Guo, N. Sonboli, R. Burke, and S. Zhang, "Personalized fairness-aware re-ranking for microlending," in *Proceedings of the 13th ACM Conference on Recommender Systems*, pp. 467–471, 2019.

[33] P. Adamopoulos and A. Tuzhilin, "The business value of recommendations: A privacy-preserving econometric analysis," in *36th International Conference on Information Systems: ICIS 2015*, Association for Information Systems, 2015.

[34] C. Pei, X. Yang, Q. Cui, X. Lin, F. Sun, P. Jiang, W. Ou, and Y. Zhang, "Value-aware recommendation based on reinforcement profit maximization," in *The World Wide Web Conference*, pp. 3123–3129, 2019.

[35] H. Abdollahpouri, R. Burke, and B. Mobasher, "Value-aware item weighting for long-tail recommendation," *arXiv preprint arXiv:1802.05382*, 2018.

[36] R. Louca, M. Bhattacharya, D. Hu, and L. Hong, "Joint optimization of profit and relevance for recommendation systems in e-commerce.," in *RMSE@ RecSys*, 2019.

[37] D. Jannach and G. Adomavicius, "Price and profit awareness in recommender systems," *arXiv preprint arXiv:1707.08029*, 2017.

[38] Y. Li, Y. Zhang, L. Gan, G. Hong, Z. Zhou, and Q. Li, "Revman: Revenue-aware multi-task online insurance recommendation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 303–310, 2021.

[39] R. Yang, M. Xu, P. Jones, and N. Samatova, "Real time utility-based recommendation for revenue optimization via an adaptive online top-k high utility itemsets mining model," in *2017 13th international conference on natural computation, fuzzy systems and knowledge discovery (ICNC-FSKD)*, pp. 1859–1866, IEEE, 2017.

[40] P. Hosein, I. Rahaman, K. Nichols, and K. Maharaj, "Recommendations for long-term profit optimization.," in *ImpactRS@ RecSys*, 2019.

[41] N. Ghanem, S. Leitner, and D. Jannach, "Balancing consumer and business value of recommender systems: A simulation-based analysis," *arXiv preprint arXiv:2203.05952*, 2022.

[42] K. Hosanagar, R. Krishnan, and L. Ma, "Recomended for you: The impact of profit incentives on the relevance of online recommendations," *ICIS 2008 Proceedings*, p. 31, 2008.

[43] U. Panniello, S. Hill, and M. Gorgoglione, "The impact of profit incentives on the relevance of online recommendations," *Electronic Commerce Research and Applications*, vol. 20, pp. 87–104, 2016.

[44] A. Das, C. Mathieu, and D. Ricketts, "Maximizing profit using recommender systems," *arXiv preprint arXiv:0908.3633*, 2009.

[45] G. Shani, D. Heckerman, R. I. Brafman, and C. Boutilier, "An mdp-based recommender system.," *Journal of Machine Learning Research*, vol. 6, no. 9, 2005.

[46] Y. Cai and D. Zhu, "Trustworthy and profit: A new value-based neighbor selection method in recommender systems under shilling attacks," *Decision Support Systems*, vol. 124, p. 113112, 2019.

[47] M. Brand, "A random walks perspective on maximizing satisfaction and profit," in *Proceedings of the 2005 SIAM international conference on data mining*, pp. 12–19, SIAM, 2005.

[48] L. Akoglu and C. Faloutsos, "Valuepick: Towards a value-oriented dual-goal recommender system," in *2010 IEEE International Conference on Data Mining Workshops*, pp. 1151–1158, IEEE, 2010.

[49] M. Kompan, P. Gaspar, J. Macina, M. Cimerman, and M. Bielikova, "Exploring customer price preference and product profit role in recommender systems," *IEEE Intelligent Systems*, vol. 37, no. 1, pp. 89–98, 2021.

[50] L.-S. Chen, F.-H. Hsu, M.-C. Chen, and Y.-C. Hsu, "Developing recommender systems with the consideration of product profitability for sellers," *Information Sciences*, vol. 178, no. 4, pp. 1032–1048, 2008.

[51] I. Benouaret, S. Amer-Yahia, C. Kamdem-Kengne, and J. Chagraoui, "A bi-objective approach for product recommendations," in *2019 IEEE International Conference on Big Data (Big Data)*, pp. 2159–2168, IEEE, 2019.

[52] C. Long, R. C.-W. Wong, and V. J. Wei, "Profit maximization with sufficient customer satisfactions," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 12, no. 2, pp. 1–34, 2018.

[53] V. Kini and A. Manjunatha, "Revenue maximization using multitask learning for promotion recommendation," in *2020 International Conference on Data Mining Workshops (ICDMW)*, pp. 144–150, IEEE, 2020.

[54] R. Ma, H. Li, J. Cen, and A. Arora, "Placement-and-profit-aware association rules mining.," in *ICAART (2)*, pp. 639–646, 2019.

[55] Y. Nemati and H. Khademolhosseini, "Devising a profit-aware recommender system using multi-objective ga," *Journal of Advances in Computer Research*, vol. 11, no. 3, pp. 109–120, 2020.

[56] A. Azaria, A. Hassidim, S. Kraus, A. Eshkol, O. Weintraub, and I. Netanely, "Movie recommender system for profit maximization," in *Proceedings of the 7th ACM conference on Recommender systems*, pp. 121–128, 2013.

[57] H.-F. Wang and C.-T. Wu, "A mathematical model for product selection strategies in a recommender system," *Expert Systems with Applications*, vol. 36, no. 3, pp. 7299–7308, 2009.

[58] D. Lee, K. Nam, I. Han, and K. Cho, "From free to fee: Monetizing digital content through expected utility-based recommender systems," *Information & Management*, vol. 59, no. 6, p. 103681, 2022.

[59] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[60] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*, pp. 242–264, IGI global, 2010.

[61] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, pp. 1–40, 2016.

[62] M. E. Taylor and P. Stone, "Transfer learning for reinforcement learning domains: A survey.," *Journal of Machine Learning Research*, vol. 10, no. 7, 2009.

[63] G. A. Rummery and M. Niranjan, *On-line Q-learning using connectionist systems*, vol. 37. Citeseer, 1994.

[64] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3, pp. 279–292, 1992.

[65] F. L. Da Silva, G. Warnell, A. H. R. Costa, and P. Stone, "Agents teaching agents: a survey on inter-agent transfer learning," *Autonomous Agents and Multi-Agent Systems*, vol. 34, no. 1, pp. 1–17, 2020.

[66] S. Schaal, "Learning from demonstration," *Advances in neural information processing systems*, vol. 9, 1996.

[67] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, "Recent advances in robot learning from demonstration," *Annual review of control, robotics, and autonomous systems*, vol. 3, pp. 297–330, 2020.

[68] W. B. Knox and P. Stone, "Interactively shaping agents via human reinforcement: The tamer framework," in *Proceedings of the fifth international conference on Knowledge capture*, pp. 9–16, 2009.

[69] A. Y. Ng, D. Harada, and S. Russell, "Policy invariance under reward transformations: Theory and application to reward shaping," in *Icml*, vol. 99, pp. 278–287, 1999.

[70] A. Laud and G. DeJong, "The influence of reward on the speed of reinforcement learning: An analysis of shaping," in *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pp. 440–447, 2003.

[71] L. Torrey and M. Taylor, "Teaching on a budget: Agents advising agents in reinforcement learning," in *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pp. 1053–1060, 2013.

[72] G. Hinton, O. Vinyals, J. Dean, *et al.*, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, vol. 2, no. 7, 2015.

[73] C. Buciluǎ, R. Caruana, and A. Niculescu-Mizil, "Model compression," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 535–541, 2006.

[74] A. A. Rusu, S. G. Colmenarejo, C. Gulcehre, G. Desjardins, J. Kirkpatrick, R. Pascanu, V. Mnih, K. Kavukcuoglu, and R. Hadsell, "Policy distillation," *arXiv preprint arXiv:1511.06295*, 2015.

[75] Y. Zhan, A. Fachantidis, I. Vlahavas, and M. E. Taylor, "Agents teaching humans in reinforcement learning tasks," in *Proceedings of the Adaptive and Learning Agents Workshop (AAMAS)*, 2014.

[76] A. Fachantidis, M. E. Taylor, and I. Vlahavas, "Learning to teach reinforcement learning agents," *Machine Learning and Knowledge Extraction*, vol. 1, no. 1, pp. 21–42, 2017.

[77] E. Ilhan, J. Gow, and D. Perez, "Student-initiated action advising via advice novelty," *IEEE Transactions on Games*, 2021.

[78] Y. Spielberg and A. Azaria, "Criticality-based advice in reinforcement learning," in *Proceedings of the Annual Meeting of the Cognitive Science Society*, vol. 44, 2022.

[79] M. Zimmer, P. Viappiani, and P. Weng, "Teacher-student framework: a reinforcement learning approach," in *AAMAS Workshop Autonomous Robots and Multirobot Systems*, 2014.

[80] J. A. Clouse, *On integrating apprentice learning and reinforcement learning*. University of Massachusetts Amherst, 1996.

[81] Y. Guo, R. Li, J. Campbell, D. Hughes, F. Fang, and K. Sycara, "A teacher-student policy transfer framework for giving explainable action advice in multi-agent reinforcement learning,"

[82] D. Anand, V. Gupta, P. Paruchuri, and B. Ravindran, "An enhanced advising model in teacher-student framework using state categorization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 6653–6660, 2021.

[83] D.-K. Kim, M. Liu, S. Omidshafiei, S. Lopez-Cot, M. Riemer, G. Habibi, G. Tesauro, S. Mourad, M. Campbell, and J. P. How, "Learning hierarchical teaching in cooperative multiagent reinforcement learning," 2019.

[84] F. Cruz, S. Magg, Y. Nagai, and S. Wermter, "Improving interactive reinforcement learning: What makes a good teacher?," *Connection Science*, vol. 30, no. 3, pp. 306–325, 2018.

[85] E. Ilhan, J. Gow, and D. Perez-Liebana, "Teaching on a budget in multi-agent deep reinforcement learning," in *2019 IEEE Conference on Games (CoG)*, pp. 1–8, IEEE, 2019.

[86] S. Omidshafiei, D.-K. Kim, M. Liu, G. Tesauro, M. Riemer, C. Amato, M. Campbell, and J. P. How, "Learning to teach in cooperative multiagent reinforcement learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 6128–6136, 2019.

[87] F. Cruz, P. Wüppen, S. Magg, A. Fazrie, and S. Wermter, "Agent-advising approaches in an interactive reinforcement learning scenario," in *2017 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pp. 209–214, IEEE, 2017.

[88] M. E. Taylor, N. Carboni, A. Fachantidis, I. Vlahavas, and L. Torrey, "Reinforcement learning agents providing advice in complex video games," *Connection Science*, vol. 26, no. 1, pp. 45–63, 2014.

[89] C. Zhu, Y. Cai, H.-f. Leung, and S. Hu, "Learning by reusing previous advice in teacher-student paradigm," in *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 1674–1682, 2020.

[90] E. Ilhan, J. Gow, and D. Perez-Liebana, "Learning on a budget via teacher imitation," in *2021 IEEE Conference on Games (CoG)*, pp. 1–8, IEEE, 2021.

[91] E. Ilhan, J. Gow, and D. Perez-Liebana, "Action advising with advice imitation in deep reinforcement learning," *arXiv preprint arXiv:2104.08441*, 2021.

[92] F. L. Da Silva, R. Glatt, and A. H. R. Costa, "Simultaneously learning and advising in multiagent reinforcement learning," in *Proceedings of the 16th conference on autonomous agents and multiagent systems*, pp. 1100–1108, 2017.

[93] F. L. Da Silva, P. Hernandez-Leal, B. Kartal, and M. E. Taylor, "Uncertainty-aware action advising for deep reinforcement learning agents," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, pp. 5792–5799, 2020.

[94] O. Amir, E. Kamar, A. Kolobov, and B. Grosz, "Interactive teaching strategies for agent training," in *In Proceedings of IJCAI 2016*, 2016.

[95] S. P. Singh and R. S. Sutton, "Reinforcement learning with replacing eligibility traces," *Machine learning*, vol. 22, no. 1, pp. 123–158, 1996.

[96] S. Mahadevan, "To discount or not to discount in reinforcement learning: A case study comparing r learning and q learning," in *Machine Learning Proceedings 1994*, pp. 164–172, Elsevier, 1994.

[97] E. Fry, "Teaching machine dichotomy: Skinner vs. pressey," *Psychological Reports*, vol. 6, no. 1, pp. 11–14, 1960.

[98] J. R. Anderson, C. F. Boyle, and B. J. Reiser, "Intelligent tutoring systems," *Science*, vol. 228, no. 4698, pp. 456–462, 1985.

[99] B. J. Fogg, "Persuasive computers: perspectives and research directions," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 225–232, 1998.

[100] B. J. Fogg, "Persuasive technology: using computers to change what we think and do," *Ubiquity*, vol. 2002, no. December, p. 2, 2002.

[101] S. B. Drewe, "An examination of the relationship between coaching and teaching," *Quest*, vol. 52, no. 1, pp. 79–88, 2000.

[102] H. Oinas-Kukkonen, "A foundation for the study of behavior change support systems," *Personal and ubiquitous computing*, vol. 17, no. 6, pp. 1223–1235, 2013.

[103] R. Orji and K. Moffatt, "Persuasive technology for health and wellness: State-of-the-art and emerging trends," *Health informatics journal*, vol. 24, no. 1, pp. 66–91, 2018.

[104] C. Pinder, J. Vermeulen, B. R. Cowan, and R. Beale, "Digital behaviour change interventions to break and form habits," *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 25, no. 3, pp. 1–66, 2018.

[105] J. Hamari, J. Koivisto, and T. Pakkanen, "Do persuasive technologies persuade?-a review of empirical studies," in *International conference on persuasive technology*, pp. 118–136, Springer, 2014.

[106] I. Wiafe and K. Nakata, "Bibliographic analysis of persuasive systems: techniques; methods and domains of application," in *Persuasive technology: Design for health and safety; the 7th international conference on persuasive technology; PERSUASIVE 2012; Linköping; Sweden; June 6-8; Adjunct Proceedings*, no. 068, pp. 61–64, Linköping University Electronic Press, 2012.

[107] I. Adaji and M. Adisa, "A review of the use of persuasive technologies to influence sustainable behaviour," in *Adjunct Proceedings of the 30th ACM Conference on User Modeling, Adaptation and Personalization*, pp. 317–325, 2022.

[108] K. Torning and H. Oinas-Kukkonen, "Persuasive system design: state of the art and future directions," in *Proceedings of the 4th international conference on persuasive technology*, pp. 1–8, 2009.

[109] L. Anselma and A. Mazzei, "Building a persuasive virtual dietitian," in *Informatics*, vol. 7, p. 27, MDPI, 2020.

[110] B. J. Fogg, "A behavior model for persuasive design," in *Proceedings of the 4th international Conference on Persuasive Technology*, pp. 1–7, 2009.

[111] C. Conati, "Intelligent tutoring systems: New challenges and directions," in *Twenty-First International Joint Conference on Artificial Intelligence*, 2009.

[112] H. S. Nwana, "Intelligent tutoring systems: an overview," *Artificial Intelligence Review*, vol. 4, no. 4, pp. 251–277, 1990.

[113] A. T. Corbett, K. R. Koedinger, and J. R. Anderson, "Intelligent tutoring systems," in *Handbook of human-computer interaction*, pp. 849–874, Elsevier, 1997.

[114] D. M. Towne and A. Munro, "Supporting diverse instructional strategies in a simulation-oriented training environment," in *Cognitive approaches to automated instruction*, pp. 107–134, Routledge, 2013.

[115] J. Vandeputte, A. Cornuéjols, N. N. Darcel, F. Delaere, and C. Martin, "Le coaching: un nouveau cadre pour la recommandation automatique en vue de modifications durables du comportement," in *CNIA 2021: Conférence Nationale en Intelligence Artificielle*, pp. 44–51, 2021.

[116] J. Vandeputte, A. Cornuéjols, N. Darcel, F. Delaere, and C. Martin, "Coaching agent: Making recommendations for behavior change. a case study on improving eating habits," in *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, pp. 1292–1300, 2022.

[117] M. M. Afsar, T. Crump, and B. Far, "Reinforcement learning based recommender systems: A survey," *arXiv preprint arXiv:2101.06286*, 2021.

[118] M. Chen, A. Beutel, P. Covington, S. Jain, F. Belletti, and E. H. Chi, "Top-k off-policy correction for a reinforce recommender system," in *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pp. 456–464, 2019.

[119] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the 10th international conference on World Wide Web*, pp. 285–295, 2001.

[120] M. Rayner, "Nutrient profiling for regulatory purposes," *Proceedings of the Nutrition Society*, vol. 76, no. 3, pp. 230–236, 2017.

[121] S. Akkoyunlu, C. Manfredotti, A. Cornuéjols, N. Darcel, and F. Delaere, "Investigating substitutability of food items in consumption data," in *Second International Workshop on Health Recommender Systems co-located with ACM RecSys*, vol. 5, 2017.

[122] W. Willett, J. Rockström, B. Loken, M. Springmann, T. Lang, S. Vermeulen, T. Garnett, D. Tilman, F. DeClerck, A. Wood, *et al.*, "Food in the anthropocene: the eat–lancet commission on healthy diets from sustainable food systems," *The Lancet*, vol. 393, no. 10170, pp. 447–492, 2019.

[123] M. M. Jastran, C. A. Bisogni, J. Sobal, C. Blake, and C. M. Devine, "Eating routines. embedded, value based, modifiable, and reflective," *Appetite*, vol. 52, no. 1, pp. 127–136, 2009.

[124] "Food.com." https://www.food.com/. Accessed: 2022-09-30.

[125] "Allrecipes.com." https://www.allrecipes.com. Accessed: 2022-09-30.

[126] "Epicurious.com." https://www.epicurious.com. Accessed: 2022-09-30.

[127] J. Wardle, A. M. Haase, A. Steptoe, M. Nillapun, K. Jonwutiwes, and F. Bellisie, "Gender differences in food choice: the contribution of health beliefs and dieting," *Annals of behavioral medicine*, vol. 27, no. 2, pp. 107–116, 2004.

[128] C. A. Bisogni, M. Jastran, M. Seligson, and A. Thompson, "How people interpret healthy eating: contributions of qualitative research," *Journal of nutrition education and behavior*, vol. 44, no. 4, pp. 282–301, 2012.

[129] Elsevier, "Scopus." https://www.elsevier.com/solutions/scopus/content. Accessed: 2022-10-25.

[130] B. Pathak, R. Garfinkel, R. D. Gopal, R. Venkatesan, and F. Yin, "Empirical analysis of the impact of recommender systems on sales," *Journal of Management Information Systems*, vol. 27, no. 2, pp. 159–188, 2010.

[131] J. Kamahara, T. Asakawa, S. Shimojo, and H. Miyahara, "A community-based recommendation system to reveal unexpected interests," in *11th international multimedia modelling conference*, pp. 433–438, IEEE, 2005.

[132] J. Freyne and S. Berkovsky, "Intelligent food planning: personalized recipe recommendation," in *Proceedings of the 15th international conference on Intelligent user interfaces*, pp. 321–324, 2010.

[133] H. Jung and K. Chung, "Knowledge-based dietary nutrition recommendation for obese management," *Information Technology and Management*, vol. 17, no. 1, pp. 29–42, 2016.

[134] M. Harvey, B. Ludwig, and D. Elsweiler, "You are what you eat: Learning user tastes for rating prediction," in *International symposium on string processing and information retrieval*, pp. 153–164, Springer, 2013.

[135] M. Ueda, M. Takahata, and S. Nakajima, "User's food preference extraction for personalized cooking recipe recommendation," in *Workshop of ISWC*, pp. 98–105, 2011.

[136] C.-S. Lee, M.-H. Wang, and H. Hagras, "A type-2 fuzzy ontology and its application to personal diabetic-diet recommendation," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 2, pp. 374–395, 2010.

[137] M. Khan, "A topic modelling based approach towards personalized and health-aware food recommendation," 2022.

[138] C. Celis-Morales, K. M. Livingstone, C. F. Marsaux, H. Forster, C. B. O'Donovan, C. Woolhead, A. L. Macready, R. Fallaize, S. Navas-Carretero, R. San-Cristobal, *et al.*, "Design and baseline characteristics of the food4me study: a web-based randomised controlled trial of personalised nutrition in seven european countries," *Genes & nutrition*, vol. 10, no. 1, pp. 1–13, 2015.

[139] Y. Van Pinxteren, G. Geleijnse, and P. Kamsteeg, "Deriving a recipe similarity measure for recommending healthful meals," in *Proceedings of the 16th international conference on Intelligent user interfaces*, pp. 105–114, 2011.

[140] F. Cordeiro, E. Bales, E. Cherry, and J. Fogarty, "Rethinking the mobile food journal: Exploring opportunities for lightweight photo-based capture," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 3207–3216, 2015.

[141] L. Yang, C.-K. Hsieh, H. Yang, J. P. Pollak, N. Dell, S. Belongie, C. Cole, and D. Estrin, "Yum-me: a personalized nutrient-based meal recommender system," *ACM Transactions on Information Systems (TOIS)*, vol. 36, no. 1, pp. 1–31, 2017.

[142] C. K. Martin, T. Nicklas, B. Gunturk, J. B. Correa, H. R. Allen, and C. Champagne, "Measuring food intake with digital photography," *Journal of Human Nutrition and Dietetics*, vol. 27, pp. 72–81, 2014.

[143] S. Valtolina, M. Mesiti, and B. Barricelli, "User-centered recommendation services in internet of things era," in *CoPDA2014 workshop. Como, Italy*, 2014.

[144] J. Aberg, "Dealing with malnutrition: A meal planning system for elderly.," in *AAAI spring symposium: argumentation for consumers of healthcare*, pp. 1–7, 2006.

[145] J. Caldeira, R. S. Oliveira, L. Marinho, and C. Trattner, "Healthy menus recommendation: optimizing the use of the pantry," in *Proceedings of the 3rd International Workshop on Health Recommender Systems Co-Located with ACM RecSys*, 2018.

[146] F. Pecune, L. Callebert, and S. Marsella, "A recommender system for healthy and personalized recipes recommendations.," in *HealthRecSys@ RecSys*, pp. 15–20, 2020.

[147] P. Chavan, B. Thoms, and J. Isaacs, "A recommender system for healthy food choices: building a hybrid model for recipe recommendations using big data sets," 2021.

[148] D. Elsweiler, C. Trattner, and M. Harvey, "Exploiting food choice biases for healthier recipe recommendation," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, (New York, NY, USA), p. 575–584, Association for Computing Machinery, 2017.

[149] M. Ge, F. Ricci, and D. Massimo, "Health-aware food recommender system," in *Proceedings of the 9th ACM Conference on Recommender Systems*, pp. 333–334, 2015.

[150] F. Abbas, N. Najjar, and D. Wilson, "Increasing diversity through dynamic critique in conversational recipe recommendations," in *Proceedings of the 13th International Workshop on Multimedia for Cooking and Eating Activities*, pp. 9–16, 2021.

[151] J. Loesch, L. Meeckers, I. van Lier, A. de Boer, M. Dumontier, and R. Celebi, "Automated identification of food substitutions using knowledge graph embeddings.," in *SWAT4HCLS*, pp. 19–28, 2022.

[152] P. Achananuparp and I. Weber, "Extracting food substitutes from food diary via distributional similarity," *arXiv preprint arXiv:1607.08807*, 2016.

[153] J. Zhang, Y. J. Oh, P. Lange, Z. Yu, Y. Fukuoka, *et al.*, "Artificial intelligence chatbot behavior change model for designing artificial intelligence chatbots to promote physical activity and a healthy diet," *Journal of medical Internet research*, vol. 22, no. 9, p. e22845, 2020.

[154] F. Pecune, L. Callebert, and S. Marsella, "Designing persuasive food conversational recommender systems with nudging and socially-aware conversational strategies," *Frontiers in Robotics and AI*, p. 390, 2021.

[155] G. Castiglia, A. El Majjodi, F. Calò, Y. Deldjoo, F. Narducci, A. Starke, and C. Trattner, "Nudging towards health in a conversational food recommender system using multi-modal interactions and nutrition labels," 2022.

[156] P. K. Prasetyo, P. Achananuparp, and E.-P. Lim, "Foodbot: A goal-oriented just-in-time healthy eating interventions chatbot," in *Proceedings of the 14th EAI International Conference on Pervasive Computing Technologies for Healthcare*, pp. 436–439, 2020.

[157] T. N. Trang Tran, M. Atas, A. Felfernig, and M. Stettinger, "An overview of recommender systems in the healthy food domain," *Journal of Intelligent Information Systems*, vol. 50, no. 3, pp. 501–526, 2018.

[158] L. Hagen, "Pretty healthy food: How and when aesthetics enhance perceived healthiness," *Journal of Marketing*, vol. 85, no. 2, pp. 129–145, 2021.

[159] B. Lika, K. Kolomvatsos, and S. Hadjiefthymiades, "Facing the cold start problem in recommender systems," *Expert systems with applications*, vol. 41, no. 4, pp. 2065–2073, 2014.

[160] Anses, "Ciqual Database." `https://ciqual.anses.fr/`. Accessed: 2022-10-25.

[161] Z. J. Schwarz F, "PennController for Internet Based Experiments (IBEX)." `https://www.pcibex.net/`. Accessed: 2022-10-25.

[162] A. Fadhil, "Can a chatbot determine my diet?: Addressing challenges of chatbot application for meal recommendation," *arXiv preprint arXiv:1802.09100*, 2018.

[163] L. Chen and P. Pu, "Critiquing-based recommenders: survey and emerging trends," *User Modeling and User-Adapted Interaction*, vol. 22, no. 1, pp. 125–150, 2012.

[164] B. Loepp, T. Hussein, and J. Ziegler, "Choice-based preference elicitation for collaborative filtering recommender systems," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 3085–3094, 2014.

[165] M. I. Norton, D. Mochon, and D. Ariely, "The ikea effect: When labor leads to love," *Journal of consumer psychology*, vol. 22, no. 3, pp. 453–460, 2012.

[166] C. Sardianos, I. Varlamis, C. Chronis, G. Dimitrakopoulos, A. Alsalemi, Y. Himeur, F. Bensaali, and A. Amira, "The emergence of explainability of intelligent systems: Delivering explainable and personalized recommendations for energy efficiency," *International Journal of Intelligent Systems*, vol. 36, no. 2, pp. 656–680, 2021.

[167] V. L. Fulgoni III, D. R. Keast, and A. Drewnowski, "Development and validation of the nutrient-rich foods index: a tool to measure nutritional quality of foods," *The Journal of nutrition*, vol. 139, no. 8, pp. 1549–1554, 2009.

[168] M.-È. Labonté, T. Poon, B. Gladanac, M. Ahmed, B. Franco-Arellano, M. Rayner, and M. R. L'Abbé, "Nutrient profile models with applications in government-led nutrition policies aimed at health promotion and noncommunicable disease prevention: a systematic review," *Advances in Nutrition*, vol. 9, no. 6, pp. 741–788, 2018.

[169] A. Drewnowski and V. Fulgoni III, "Nutrient profiling of foods: creating a nutrient-rich food index," *Nutrition reviews*, vol. 66, no. 1, pp. 23–39, 2008.

[170] J. Chantal, S. Hercberg, W. H. Organization, *et al.*, "Development of a new front-of-pack nutrition label in france: the five-colour nutri-score," *Public health panorama*, vol. 3, no. 04, pp. 712–725, 2017.

[171] K. Zelman and E. Kennedy, "Naturally nutrient rich... putting more power on americans' plates," *Nutrition Today*, vol. 40, no. 2, pp. 60–68, 2005.

[172] D. Chaltiel, M. Adjibade, V. Deschamps, M. Touvier, S. Hercberg, C. Julia, and E. Kesse-Guyot, "Programme national nutrition santé–guidelines score 2 (pnns-gs2): development and validation of a diet quality score reflecting the 2017 french dietary guidelines," *British Journal of Nutrition*, vol. 122, no. 3, pp. 331–342, 2019.

[173] S. Taechangam, U. Pinitchun, and C. Pachotikarn, "Development of nutrition education tool: healthy eating index in thailand," *Asia Pac J Clin Nutr*, vol. 17, no. Suppl 1, pp. 365–7, 2008.

[174] E. T KENNEDY, J. Ohls, S. Carlson, and K. Fleming, "The healthy eating index: design and applications," *Journal of the American dietetic association*, vol. 95, no. 10, pp. 1103–1108, 1995.

[175] B. Shatenstein, S. Nadon, C. Godin, and G. Ferland, "Diet quality of montreal-area adults needs improvement: estimates from a self-administered food frequency questionnaire furnishing a dietary indicator score," *Journal of the American Dietetic Association*, vol. 105, no. 8, pp. 1251–1260, 2005.

[176] P. Huijbregts, E. Feskens, L. Räsänen, F. Fidanza, A. Nissinen, A. Menotti, and D. Kromhout, "Dietary pattern and 20 year mortality in elderly men in finland, italy, and the netherlands: longitudinal cohort study," *Bmj*, vol. 315, no. 7099, pp. 13–17, 1997.

[177] D. B. Panagiotakos, C. Pitsavos, and C. Stefanadis, "Dietary patterns: a mediterranean diet score and its relation to clinical and biological markers of cardiovascular disease risk," *Nutrition, Metabolism and Cardiovascular Diseases*, vol. 16, no. 8, pp. 559–568, 2006.

[178] D. Chaltiel, V. Deschamps, M. Touvier, S. Hercberg, C. Julia, and E. Kesse-Guyot, "Analyse de l'association prospective entre le score d'adéquation aux recommandations nutritionnelles françaises de 2017 (pnns-gs2) et l'apparition de surpoids et d'obésité dans la cohorte nutrinet-santé," *Nutrition Clinique et Métabolisme*, vol. 33, no. 1, p. 106, 2019.

[179] C. Estaquio, E. Kesse-Guyot, V. Deschamps, S. Bertrais, L. Dauchet, P. Galan, S. Hercberg, and K. Castetbon, "Adherence to the french programme national nutrition sante guideline score is associated with better nutrient intake and nutritional status," *Journal of the American Dietetic Association*, vol. 109, no. 6, pp. 1031–1041, 2009.

[180] G. Kourlaba and D. B. Panagiotakos, "Dietary quality indices and human health: A review," *Maturitas*, vol. 62, no. 1, pp. 1–8, 2009.

[181] H. A. Guthrie, K. Black, and J. P. Madden, "Nutritional practices of elderly citizens in rural pennsylvania," *The Gerontologist*, vol. 12, no. 4, pp. 330–335, 1972.

[182] E. O. Verger, F. Mariotti, B. A. Holmes, D. Paineau, and J.-F. Huneau, "Evaluation of a diet quality index based on the probability of adequate nutrient intake (pandiet) using national french and us dietary surveys," 2012.

[183] L. A. Suchman, *Plans and situated actions: The problem of human-machine communication*. Cambridge university press, 1987.

[184] S. M. Smith, "Remembering in and out of context.," *Journal of Experimental Psychology: Human Learning and Memory*, vol. 5, no. 5, p. 460, 1979.

[185] K. A. Klein, R. M. Shiffrin, and A. H. Criss, "Putting context in context.," 2007.

[186] A. K. Dey, "Understanding and using context," *Personal and ubiquitous computing*, vol. 5, no. 1, pp. 4–7, 2001.

[187] P. Dourish, "What we talk about when we talk about context," *Personal and ubiquitous computing*, vol. 8, no. 1, pp. 19–30, 2004.

[188] G. Adomavicius and A. Tuzhilin, "Context-aware recommender systems," in *Recommender systems handbook*, pp. 217–253, Springer, 2011.

[189] K. Bauman and A. Tuzhilin, "Know thy context: parsing contextual information from user reviews for recommendation purposes," *Information Systems Research*, vol. 33, no. 1, pp. 179–202, 2022.

[190] N. Hariri, B. Mobasher, R. Burke, and Y. Zheng, "Context-aware recommendation based on review mining," in *ITWP@ IJCAI*, 2011.

[191] S. Wang, L. Hu, Y. Wang, L. Cao, Q. Z. Sheng, and M. Orgun, "Sequential recommender systems: challenges, progress and prospects," *arXiv preprint arXiv:2001.04830*, 2019.

[192] M. Quadrana, P. Cremonesi, and D. Jannach, "Sequence-aware recommender systems," *ACM Computing Surveys (CSUR)*, vol. 51, no. 4, pp. 1–36, 2018.

[193] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 1789–1819, 2021.

[194] P. Mateti and N. Deo, "On algorithms for enumerating all circuits of a graph," *SIAM Journal on Computing*, vol. 5, no. 1, pp. 90–99, 1976.

[195] D. B. Johnson, "Finding all the elementary circuits of a directed graph," *SIAM Journal on Computing*, vol. 4, no. 1, pp. 77–84, 1975.

[196] J. L. Szwarcfiter and P. E. Lauer, "A search strategy for the elementary cycles of a directed graph," *BIT Numerical Mathematics*, vol. 16, no. 2, pp. 192–204, 1976.

[197] S. S. Anand and B. Mobasher, "Contextual recommendation," in *Workshop on Web Mining*, pp. 142–160, Springer, 2006.

[198] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under concept drift: A review," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 12, pp. 2346–2363, 2018.

[199] J. J. Williams and T. Lombrozo, "Explanation and prior knowledge interact to guide learning," *Cognitive psychology*, vol. 66, no. 1, pp. 55–84, 2013.

[200] N. Tintarev, "Explanations of recommendations," in *Proceedings of the 2007 ACM conference on Recommender systems*, pp. 203–206, 2007.

[201] Y. Zhang, X. Chen, *et al.*, "Explainable recommendation: A survey and new perspectives," *Foundations and Trends® in Information Retrieval*, vol. 14, no. 1, pp. 1–101, 2020.

[202] A. H. Nabizadeh, J. P. Leal, H. N. Rafsanjani, and R. R. Shah, "Learning path personalization and recommendation methods: A survey of the state-of-the-art," *Expert Systems with Applications*, vol. 159, p. 113596, 2020.

[203] A. A. Kardan, M. A. Ebrahim, and M. B. Imani, "A new personalized learning path generation method: Aco-map," *Indian Journal of Scientific Research*, vol. 5, no. 1, pp. 17–24, 2014.

[204] M. A. Morris, C. Hulme, G. P. Clarke, K. L. Edwards, and J. E. Cade, "What is the cost of a healthy diet? using diet data from the uk women's cohort study," *J Epidemiol Community Health*, vol. 68, no. 11, pp. 1043–1049, 2014.

[205] A. Håkansson, "Has it become increasingly expensive to follow a nutritious diet? insights from a new price index for nutritious diets in sweden 1980–2012," *Food & Nutrition Research*, vol. 59, no. 1, p. 26932, 2015.