Titre: Title:	Le filtrage basé sur le contenu pour la recommandation de cours (FCRC)
Auteur: Author:	Waad Gasmi
Date:	2011
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Gasmi, W. (2011). Le filtrage basé sur le contenu pour la recommandation de cours (FCRC) [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/776/

UNIVERSITÉ DE MONTRÉAL

LE FILTRAGE BASÉ SUR LE CONTENU POUR LA RECOMMANDATION DE COURS (FCRC)

WAAD GASMI

DÉPARTEMENT DE GÉNIE INFORMATIQUE ET GÉNIE LOGICIEL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION

DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES

(GÉNIE INFORMATIQUE)

DÉCEMBRE 2011

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

LE FILTRAGE BASÉ SUR LE CONTENU POUR LA RECOMMANDATION DE COURS (FCRC)

présenté par : GASMI Waad

en vue de l'obtention du diplôme de : Maîtrise ès Sciences Appliquées

a été dûment accepté par le jury d'examen constitué de :

M. GAGNON Michel, Ph.D., président

M. DESMARAIS Michel C., Ph.D., membre et directeur de recherche

M. ANTONIOL Giuliano, Ph.D., membre

REMERCIEMENTS

Je souhaitais adresser mes remerciements les plus sincères à mon directeur de recherche, Michel C. Desmarais pour les conseils, les encouragements et toute l'aide qu'il m'a apporté durant ce projet de recherche.

Je tiens à remercier aussi toutes les personnes rencontrées lors des recherches effectuées et qui ont accepté de répondre à mes questions avec gentillesse.

J'exprime ma gratitude à mon père Gasmi Hassen pour sa contribution, son soutien et sa patience.

Merci à tous et à toutes

RÉSUMÉ

La recherche d'un cours sur un sujet précis dans un répertoire d'une ou de plusieurs universités peut s'avérer fastidieuse. Seulement à Montréal, on compte plusieurs milliers de cours universitaires offerts. Le problème est accentué par la multidisciplinarité de certains cours.

Les étudiants de cycle supérieur sont responsables de choisir leur plan d'études, les cours pertinents à leur domaine de recherche, mais ce n'est pas évident qu'ils puissent faire le bon choix des cours sans avoir besoin d'être guidés ou orientés.

Encore, les étudiants du premier cycle ont souvent le problème du nombre de places limité dans un groupe de cours. Avec un outil permettant d'établir la similarité entre des cours, les étudiants pourraient trouver rapidement des cours similaires à ceux qui, pour une raison ou une autre, ne sont pas disponibles à un trimestre ou pour leur plan d'étude.

À cette fin, plusieurs systèmes de filtrage ont été proposés, mais le filtrage basé sur le contenu pour la recommandation de cours, n'a jamais été abordé avant. L'objectif est de créer un système permettant d'établir la similarité entre les cours en se basant sur leurs descriptions et sur le calcul de leur distance dans un espace vectoriel <termes, documents>.

Ce mémoire présente le système FCRC (Filtrage basé Contenu pour la Recommandation de Cours) qui fournit des suggestions de cours sur la base de leur similarité sémantique.

Les résultats montrent que la mesure de similarité basée sur le coefficient de Dice fournit des recommandations relativement précises et complètes. Le cosinus permet aussi d'obtenir de bons résultats. Ces deux mesures sont les plus performantes. Nous sommes arrivés à identifier plus que cinq cours les plus similaires à l'intérieur des dix premiers résultats.

ABSTRACT

Searching for courses on a topic in a university database or listing of courses can prove difficult. Strictly in Montreal universities, the number of courses range in the thousands. The problem is exacerbated by the fact that many courses are multidisciplinary. For graduate students in particular, who should look for courses on a topic related to their research, it implies that defining their course plan can be a difficult process that requires some assistance. Even when a course that is relevant is found, it often is not offered in the right semester or it is filled to capacity. Therefore, a system that provides a means of finding courses based on their similarity would prove very useful.

A number of systems have been developed to provide course recommendations to students, but we aim to define an approach that is solely content-based, using the similarity of course descriptions. The algorithm is based on the vector-space model of the term-document matrix.

This thesis presents the FCRC approach (content-based course recommender) which offers recommendations based on course similarity measures.

Results show that the similarity measured on the Dice coefficient between document vectors offers relatively complete and precise recommendations. The cosine is also a good measure of similarity. In general, the first 5 of 10 recommendations are relevant based on this approach, and the recall rate is close to 100%.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES ANNEXES

INTRODUCTION

La grande quantité de cours offerts par les universités pose un défi important pour l'étudiant qui cherche un cours sur un sujet particulier. Aux États-Unis seulement, on compte 6 mille universités avec plus que 15M étudiants dont la plupart utilisent un logiciel de recommandation de cours pour suivre les cours qu'ils prennent.

Les moteurs de recherche courants fournissent une forme d'aide utile, mais limitée. On ne peut contraindre les résultats à des descriptions de cours seulement et la recherche par mots-clés demeure peu adaptée à moins de bien maîtriser le vocabulaire du domaine et de connaitre ainsi exactement les termes pertinents à la recherche.

L'étudiant a donc toujours besoin d'un guide, de l'orientation et de l'assistance. Les systèmes de recommandation peuvent les aider en fournissant des recommandations de cours personnalisées.

Dans la littérature il existe quelques systèmes de recommandations de cours (SRC). Ces systèmes proposent des solutions en utilisant des méthodes techniquement avancées. Nous révisons les principaux systèmes développés jusqu'ici.

L'utilisation des systèmes de recommandation est devenue une nécessité vu qu'ils permettent de fournir l'information pertinente avec moins d'effort et dans un délai de réponse satisfaisant. La majorité des SRC actuels souffre du problème de démarrage à froid et plusieurs ressources d'informations sont exigées.

Notre travail se situe dans le contexte de la recherche et le filtrage d'information, notamment dans le cadre des systèmes de recommandation de cours. Nous adoptons une approche basée sur le contenu qui évite le problème du démarrage à froid. Plus précisément, un système qui souffre du problème de démarrage à froid est un système qui ne peut pas produire des inférences pour les utilisateurs sur lesquels il n'a pas encore recueilli suffisamment d'informations.

A travers ce mémoire, on étudie l'ensemble des recommandations des cours produites par les systèmes de recommandation de cours actuels. Notre travail sera réalisé dans le cadre du développement d'un outil de recommandation FCRC (Filtrage basé sur le contenu pour la recommandation de cours). Nous travaillons avec des données réelles. Comme solution pour le problème de démarrage à froid, on propose que le filtrage des cours soit basé sur leurs descriptifs. Autrement dit, la recherche des cours similaires à la requête sera basée sur la comparaison entre

la requête et les descriptions des cours. Chaque cours est déjà représenté par sa description. Le principe de base de la recherche consiste donc à effectuer une « requête » à partir d'un exemple de cours. Cette approche est particulièrement indiquée lorsque l'étudiant désire identifier des cours équivalents, une situation très fréquente.

En effet, l'un des principaux défis que nous abordons dans notre étude est l'inutilité de certains termes trouvés dans les descriptions des cours en matière de pertinence. Dans la littérature, l'attribution des poids aux termes est une solution à ce problème.

En résumant, notre mémoire est organisé comme suit :

- Nous commençons par définir les interfaces intelligentes, les interfaces de recommandation et la notion de recherche d'informations en citant quelques stratégies de recherche telles que : les modèles de l'espace vectoriel et l'indexation sémantique. Aussi on présente les techniques de filtrage d'information : filtrage collaboratif, filtrage basé sur le contenu et filtrage hybride.
- Nous étudions une variété d'algorithmes et des techniques employées dans différents systèmes de recommandation.
- Nous présentons une revue de littérature des systèmes de recommandation de cours en récapitulant par un tableau des avantages et des inconvénients de chaque système.
- Ensuite, nous expliquons les étapes suivies pour la réalisation du projet.
- Nous présentons les résultats de l'évaluation du modèle proposé.

CHAPITRE 1 LES INTERFACES DE RECOMMANDATION

Les interfaces de recommandation sont un type d'interfaces intelligentes qui gagnent rapidement en popularité depuis plus d'une décennie. Ce chapitre présente en premier lieu les interfaces intelligentes et les domaines d'application. Par la suite, nous précisons les caractéristiques des interfaces de recommandation et décrivons dans les sections 1.3 et 1.4 quelques stratégies et méthodes de recherche et du filtrage d'informations. Finalement nous présentons quelques techniques utilisées pour les systèmes de recommandation de cours existants dans la littérature.

1.1 Les interfaces intelligentes

Le domaine des interfaces intelligentes est l'un des champs de recherche en interaction humain-machine. L'un des buts d'une interface intelligente est d'offrir une assistance interactive à l'utilisateur basée sur des techniques inspirées de l'intelligence artificielle [Waern, 1997], notamment la reconnaissance des intentions de l'utilisateur et de son profil. La naissance des premiers travaux dans ce domaine est faite dans les années 1970. Ce type d'interface peut prendre différentes formes telles que : aider l'utilisateur à trouver une information pertinente, à prévenir ou corriger des erreurs, ou encore à lui fournir des indices et de l'aide pour effectuer et compléter une tache [Bazargan, 2006]. Pour comprendre la notion d'interfaces intelligentes, nous nous référons aux travaux de Waern [Waern, 1997] qui a établi les bases conceptuelles des interfaces intelligentes.

Premièrement, il faut distinguer entre deux concepts : un «système» qui fait appel à des techniques d'intelligence artificielle n'a pas nécessairement une interface intelligente, et une interface bien conçue n'est pas nécessairement intelligente. Plus précisément, l'intelligence d'un «système intelligent» ne se manifeste pas nécessairement dans une interface. D'abord, il faut comprendre pourquoi une "bonne" interface ne peut pas être considérée comme intelligente. Actuellement on trouve plusieurs approches pour le développement d'interfaces utilisables et efficaces, guidées par des normes d'interface [Smith et Mosier 1986, Nielsen 1989] qui imposent souvent des restrictions dans le comportement de l'interface dans le but d'avoir une interface facile à apprendre. Ainsi, des interfaces très sophistiquées peuvent faciliter l'apprentissage et

fournir une assistance, mais elles ne font pas appel aux concepts de modèle utilisateur et à des techniques d'intelligence artificielle.

Il y a aussi deux définitions possibles proposées dans la littérature que Waern considère comme étant trop étroites: des systèmes qui imitent le dialogue humain, et des interfaces adaptatives. Certains chercheurs pensent qu'un système pouvant maintenir un dialogue humain est peut-être considéré comme intelligent.. Selon Wahlster et Kobsa, une interface intelligente ne signifie pas nécessairement le fait de maintenir un modèle de l'utilisateur et s'y adapter [Wahlster et Kobsa 1988]. Cependant, Waern pense que ceci est une définition possible d'interfaces intelligentes, mais elle est à éviter, car elle impose une contrainte technique. Waern considère aussi qu'il faut employer certaines techniques intelligentes pour ce type d'interface à condition qu'elle apporte de l'amélioration. L'interface résultante devrait être mieux que toute autre solution, et pas seulement différente et techniquement plus avancée. Une liste des techniques qui sont utilisées aujourd'hui est présentée dans le tableau suivant:

Tableau 1.1 Techniques employées pour les interfaces intelligentes [Waern, 1997]

Techniques	**Descriptions**
Adaptabilité à l'utilisateur	Fournit une interaction humain-machine adaptée aux différents utilisateurs et aux différentes situations d'utilisation.
La modélisation de l'utilisateur	Les techniques qui permettent de maintenir un modèle de connaissances sur un utilisateur.
La technologie du langage naturel	Permet d'interpréter ou de générer des énoncés en langue naturelle à partir d'un le texte ou d'un discours.
Modélisation du dialogue	Permet de maintenir un dialogue en langage naturel avec un utilisateur.
Explication de la génération	Permet d'expliquer ses résultats à un utilisateur.

Waern définit le domaine de recherche des interfaces intelligentes comme suit :

Le domaine de recherche d'interfaces intelligentes combine les principes de conception et les avancées technologiques pour l'interaction homme-ordinateur. La recherche sur les interfaces intelligentes vise à étendre les frontières des deux [Waern, 1997].

Une définition récente des fonctionnalités qu'une interface intelligente peut posséder est proposée dans [Sandel, 2002]. Sandel cite une vingtaine de caractéristiques qui permettent de qualifier l'intelligence d'une interface telles que l'automatisation, l'auto-adaptation comportementale, la création et la maintenance d'un modèle d'utilisateur, la suggestion de solutions spécifiques et personnalisées.

Les interfaces intelligentes sont appliquées dans différents domaines tels que : les tutoriels intelligents, aide intelligente et le filtrage de l'information. Pour cette étude, nous allons nous concentrer sur l'application des interfaces intelligentes pour le filtrage d'information. Le web actuel présente une richesse immense des informations et des connaissances qui sont représentées et organisées d'une façon non structurée. Toutefois, il est difficile pour un utilisateur de trouver les informations pertinentes. Il a certainement besoin d'aide et d'orientation pour effectuer certains choix. Cependant, le problème est exacerbé par le fait que l'utilisateur ne sait pas ce qui existe. Les techniques de filtrage et de recherche d'informations en général ont pour but de trouver la structure de l'information disponible qui peut aider les utilisateurs à naviguer dans l'espace d'information en sélectionnant les informations pertinentes. Les systèmes de recommandation sont des outils de filtrage d'information.

1.2 Les interfaces de recommandation

1.2.1 Historique

Les interfaces de recommandation facilitent l'interaction humain-machine et l'accès à l'information pertinente aux besoins de l'utilisateur. On retrace les premiers travaux portant sur le filtrage d'information à un article de Luhn [Luhn, 1958] sous le nom de « diffusion sélective d'une nouvelle information ». Le terme « filtrage d'information » a été proposé par Denning [Denning, 1982], qui s'est concentré sur le filtrage des courriers électroniques.

En 1987, deux catégories du filtrage d'information étaient proposées par Malone [Malone et coll., 1987]. La première catégorie est le filtrage cognitif nommé actuellement filtrage basé sur

le contenu. La deuxième catégorie est le filtrage social qui correspond au filtrage collaboratif (voir section 1.4).

Cet axe de recherche été très actif dans la dernière décennie et même plus tôt, et plusieurs laboratoires s'y sont consacré. Leurs travaux vont de la catégorisation du filtrage d'information au développement des techniques d'extraction des informations pour la sélection des messages. Par exemple, la conférence MUC (Message Understanding Conference) [Hirschman, 1991] rapporte une série d'études qui ont eu un impact non négligeable. Cette conférence était parrainée par l'agence DARPA (United States Advanced Research Projects Agency des États-Unis) en 1989.

Après une année, DARPA a continué les travaux sur les techniques de filtrage d'information par le lancement d'un nouveau projet nommé Tapestry qui avait pour objectif la réunion des efforts de recherche des différents participants à MUC [Harman, 1992a].

Plusieurs systèmes de recherche et de filtrage d'information ont été développés. Donc il fallait trouver un moyen pour les évaluer. DARPA collabore avec NIST (National Institut of Standards and Technology) et parrainent un projet international TREC (Text REtrieval Conference) qui concerne l'évaluation de ces systèmes [Harman, 1992b. Suite à une conférence sur l'incertitude en intelligence artificielle, une étude séminale qui compare des algorithmes prédictifs pour le filtrage collaboratif était publiée par Breese, Heckerman, et Kadie dans [Breese, Heckerman, et Kadie, 1998].

Une interface ou un système de recommandation constitue un type de filtrage d'information qui a pour objectif de prévoir et de présenter les éléments d'informations tels que les cours, la musique, les films… qui correspondront aux intérêts et besoins de l'utilisateur. Ceci peut se faire par la comparaison entre les objets eux-mêmes (approche Item-Item) ou par la comparaison entre les choix des utilisateurs (approche Utilisateur-Utilisateur). Une interface ou un système de recommandation nécessite une recherche et un filtrage d'information.

Pour recommander une liste des cours il faut les chercher d'abord en sélectionnant ceux qui sont pertinents et similaires à notre requête. Il existe plusieurs techniques et stratégies pour déterminer la similarité de documents. Elles sont expliquées dans les sections suivantes. Elles sont généralement issues de la recherche d'informations.

1.3 La recherche d'informations

Rechercher une information consiste à trouver les documents pertinents adaptés au besoin de l'utilisateur. Ce besoin est représenté par une requête qui peut être un document, une description ou des mots clés.

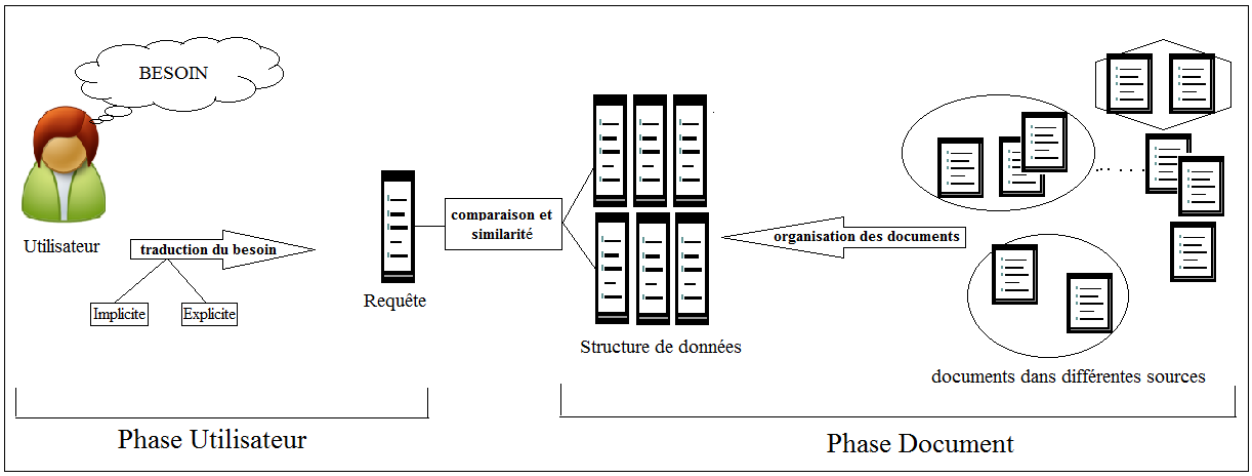


Figure 1.1 Le processus de recherche d'informations

La figure 1.1 définit la recherche d'informations comme un processus qui comporte deux phases :

-Une phase liée à l'utilisateur : cette phase décrit le besoin en information d'une personne. Si nous avons par exemple un utilisateur à la recherche d'un document D1. Ce besoin doit être traduit en une requête pour pouvoir commencer la recherche. La traduction peut être explicite quand l'utilisateur spécifie et définit en mots clés ou avec un document similaire son besoin. Mais dans le cas où nous avons un processus qui surveille le comportement de l'utilisateur, et qui permet de prédire et traduire son besoin et de préciser sa requête, nous parlons alors d'une traduction implicite.

-La deuxième phase est la phase document : supposons que nous disposons de plusieurs sources d'informations. Dans un cas pareil, il est requis de trouver un moyen pour organiser ces informations. Nous avons besoin d'une structure de données bien organisée pour que nous puissions y manipuler et générer des connaissances. Dans [Grossman et Frieder, 2004] plusieurs stratégies de recherche ont été expliquées. Nous présentons quelques-unes telles que les modèles de l'espace vectoriel et l'indexation sémantique latente dans les sections suivantes.

1.3.1 Les mesures de similarité dans l'espace vectoriel

Les mesures de similarité dans l'espace vectoriel sont le produit scalaire, le cosinus et le coefficient de Dice entre un vecteur requête et un vecteur document dans la matrice termes-documents [Salton et al., 1975]. Le modèle est basé sur l'idée que dans certains cas, le sens d'un document est exprimé par ses termes sans tenir compte la syntaxe. On y réfère parfois par l'expression « bag of words ». Si nous pouvons représenter les mots d'un document par un vecteur ça sera possible de comparer la requête avec les documents pour déterminer le degré de similarité de leurs contenus.

1.3.1.1 Le produit scalaire

Le produit scalaire entre la requête Q et un document $\boldsymbol{D}_j$ est défini comme :

$$SC(\boldsymbol{Q}, \boldsymbol{D}_j) = \sum_{\boldsymbol{i=1}}^{\boldsymbol{t}} tfq_i \times tf_{ij}$$

où :

$\boldsymbol{Q}$: le vecteur des termes de la requête

$\boldsymbol{D}_j$: vecteur des termes du document j.

tf_{ij} : la fréquence d'apparition du terme i dans le document j.

tfq_i : la fréquence du terme i dans le vecteur requête $\boldsymbol{Q}$.

Le cosinus est utilisé comme solution pour le problème trouvé avec le produit scalaire, la mesure de cosinus tient en compte la langueur du document et de la requête contrairement au produit scalaire.

1.3.1.2 Le cosinus

$$cos(\boldsymbol{Q}, \boldsymbol{D}_j) = \frac{\boldsymbol{SC}(\boldsymbol{Q}, \boldsymbol{D}_j)}{\sqrt{\sum_{j=1}^{t} tf_{ij}^2} \sqrt{\sum_{j=1}^{t} tfq_i^2}}$$

Le calcul de cosinus normalise le résultat par rapport à la longueur du document et de la requête. Alors que le produit scalaire tend à augmenter avec la longueur du document, le cosinus pénalise les documents longs en favorisant la proportion relative de termes communs. Autrement dit, plus l'angle est fermé, plus on est proche.

1.3.1.3 Le coefficient de Dice

Le coefficient de Dice est une mesure de similarité. Ce coefficient peut être défini comme deux fois l'information partagée entre un document D et une requête Q sur la somme des cardinalités.

$$S = \frac{2\,|\boldsymbol{D} \cap \boldsymbol{Q}|}{|\boldsymbol{D}| + |\boldsymbol{Q}|}$$

où

Habituellement, lorsque nous avons des vecteurs dans l'espace très proches de celui de la requête $\boldsymbol{Q}$ et ils contiennent presque les mêmes termes, les documents représentés par ces vecteurs sont les documents pertinents.

1.3.2 L'attribution des poids aux termes (fréquence inverse de document)

Pour une tâche de recherche d'information, tous les termes n'ont pas la même importance. En général, plus un terme est rare, plus il permettra d'identifier des documents spécifiques. Il s'avère donc pertinent d'attribuer un poids à chaque terme.

Le poids exprime l'importance de chaque terme. Les calculs de similarité qui reposent sur l'utilisation de poids ont donné des résultats très proches de ceux trouvés avec l'approche manuelle dans [Salton, 1969]. Le calcul du poids est basé sur l'idée qu'un terme qui n'apparaît pas fréquemment doit avoir un poids plus élevé qu'un terme qui apparaît souvent.

Les poids des termes sont calculés en utilisant la fréquence inverse du document (IDF) qui correspond à un terme donné. Le calcul de la fréquence inverse de document pour un terme t_i correspond à cette formule :

$$idf_i = \log \frac{\mathrm{N}}{\left| \{ d_j : t_i \in d_j \} \right|}$$

Où

N : est le nombre de documents

$|\{ d_j : t_i \in d_j \}|$: est le nombre de documents qui contiennent le terme t_i

Finalement, le poids s'obtient en multipliant les deux mesures :

$$tfidf_{ij} = tf_{ij} . idf_i$$

où :

tf_{ij} = la fréquence d'occurrences du terme t_i dans un document D_j

Le calcul des similarités s'effectue ainsi sur les fréquences transformées TFIDF plutôt que sur les fréquences brutes, qu'il s'agisse du cosinus ou d'autres mesures de similarité.

1.3.3 L'indexation sémantique latente

L'indexation sémantique latente (LSA) est une technique de réduction de dimension de l'espace vectoriel original de la matrice termes-documents (section 1.3.1). Elle permet d'extraire les dimensions dites latentes de cette matrice.

Les dimensions latentes représentent en quelque sorte le ou les thèmes auxquels se rapportent un document. On cherche ainsi à obtenir une appartenance des documents à ces thèmes. Il s'agit aussi d'établir l'appartenance des termes à ces mêmes thèmes et on peut alors établir les liens entre cours (ou les termes) sur la base de thèmes communs.

C'est en quelque sorte ce que la technique d'indexation sémantique latente effectue. Elle décompose la matrice termes documents en trois autres matrices :

$$\boldsymbol{M = U\,\Lambda\,V^T}$$

Où :

$\boldsymbol{M}$ **:** est la matrice termes-descriptions

$\boldsymbol{U}$ **:** est une matrice orthogonale de dimensions nombre de termes par nombre de valeurs singulières.

$\boldsymbol{\Lambda}$ **:** est une matrice diagonale des valeurs singulières

$\boldsymbol{V}$**:** est une matrice orthogonale de dimensions nombre de valeurs singulières par nombre de descriptions.

Or, en ne conservant que les valeurs singulières de $\boldsymbol{\Lambda}$ pour ne conserver que les plus importantes et fixer les autres valeurs de la diagonale à 0, appelons cette matrice $\boldsymbol{\Lambda'}$, on peut alors créer une nouvelle matrice $\boldsymbol{M'}$ où les valeurs représentent une projection dans un espace réduit de dimensions : $\boldsymbol{M' = U\,\Lambda'\,V^T}$. Le calcul de la similarité s'effectue alors sur l'appartenance des termes et des descriptions par rapport à des dimensions réduites qui peuvent en quelque sorte représenter les thèmes ou les facteurs latents [Landauer et al, 1998]. Cette technique est détaillée ci-dessous.

La méthode SVD (Décomposition des valeurs singulières)

L'occurrence d'un terme dans un document est représentée par une matrice terme-document. Cette matrice sera réduite en appliquant la méthode de décomposition des valeurs singulières SVD pour éliminer le bruit trouvé dans un document, de telle sorte que deux documents ayant la même sémantique seront localisés très proche dans un espace multidimensionnel. L'indexation sémantique permet d'éliminer les dimensions les moins pertinentes et d'effectuer une projection autour des dimensions restantes par la méthode de décomposition des valeurs singulières (SVD).

Comme mentionné, supposons une matrice terme-document nommée $\boldsymbol{M}$. Cette matrice se divise en un produit de trois matrices:

$$\boldsymbol{M = U\,\Lambda\,V^T}$$

La matrice diagonale $\boldsymbol{\Lambda}$ représente les valeurs singulières triées par leur magnitude. Ensuite, seules les premières valeurs *k* de cette matrice seront conservées, les premières *k* colonnes de $\boldsymbol{U}$ et les premières k lignes de la matrice $\boldsymbol{V^T}$. De cette façon, nous obtiendrons une nouvelle matrice, $\boldsymbol{M'}$dont on se sert pour transformer les vecteurs originaux. Les k premières valeurs singulières, peuvent alors être sélectionnées, pour obtenir $\boldsymbol{M'}$ de rang k. cette matrice représente la projection des valeurs termes et documents dans un espace a dimension réduites. Les dimensions représentent des facteurs latents auxquels les termes et les documents se rapportent. On peut ainsi dériver des corrélations entre les documents ou entre les termes sur cette base, dite sémantique.

1.3.4 Les stratégies de recherche probabiliste

Les modèles probabilistes calculent la probabilité qu'un document puisse être pertinent pour une requête. Cela réduit le problème de classement de la pertinence à une application de la théorie de probabilité. Plusieurs méthodes probabilistes sont décrites dans [Fuhr, 1992]. Deux approches fondamentales ont été proposées :

1- La première approche [Maron et Kuhns, 1920] utilise les patterns pour prédire la pertinence.

2- La deuxième approche de [Robertson and spark Jones, 1976] utilise chaque terme dans la requête comme des indices à savoir si un document est pertinent ou non.

La probabilité ici est basée sur la vraisemblance qu'un terme apparaît dans un document pertinent, elle est calculée pour chaque terme dans la collection. Pour le terme qui relie entre la requête et le document, la similarité est calculée à partir de la combinaison des probabilités de chaque terme.

$$P(A) = P(A;B) + P(A;\bar{B})$$

1.3.5 Réseau d'inférence bayésien

Pour estimer les intérêts des étudiants, le système de recommandation de cours SCR utilise un réseau bayésien qui est à la fois un modèle de représentation des connaissances et un outil pour calculer les probabilités conditionnelles afin d'inférer les documents pertinents à une requête. Ce

réseau est une description qualitative et quantitative des dépendances entre les variables, entre une requête et des documents.

Le moteur d'un réseau d'inférence utilise des relations connues pour inférer d'autres relations.

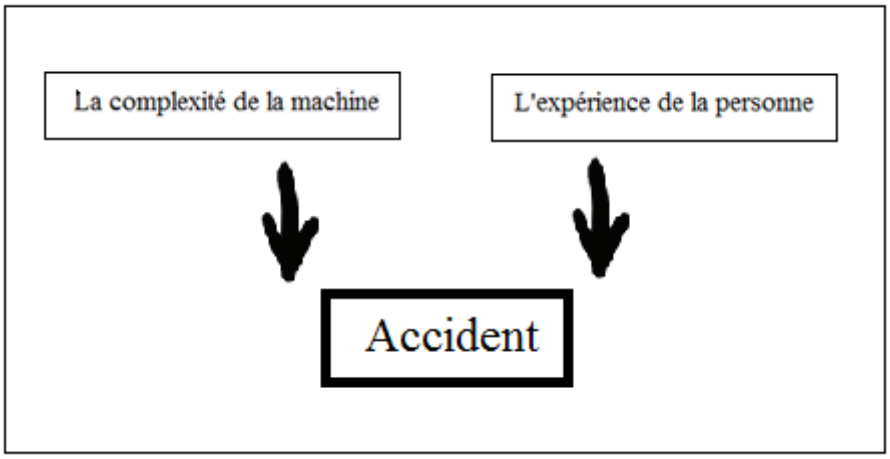


Figure 1.3 Graph de causalité.

Pour construire un réseau bayesien, il faut d'abord définir le graphe du modèle et les tables de probabilité de chaque variable, conditionnellement à ses causes.(voir figure 1.3). Une personne Y travaille sur une machine risque de tomber en panne s'il l'utilise mal. Ce risque est déterminé par deux facteurs: l'expérience de Y et la complexité de la machine. D'autres facteurs peuvent influencer le fonctionnement de la machine tels que : l'opérateur peut être fatigué, dérangé, etc.

1.3.6 La distance d'édition de Levenshtein

La distance de Levenshtein est la distance qui sépare deux mots A et B [Navarro, 2001]. Elle peut être vue comme le nombre minimal de transformations à appliquer pour passer de A à B. Une transformation peut être :

- Une suppression : on supprime une lettre du mot A.
- Une insertion : on insère une lettre dans le mot A.
- Une substitution : on change une lettre par une autre.

Prenons par exemple, les mots A et B avec A= abcdef et B= acde. Pour transformer A en B on doit effectuer deux suppressions des lettres b et f. Le système de recommandation de cours AACORN utilise cette mesure pour calculer les similarités entre les historiques des cours (2.1.3).

1.4 Le filtrage d'information

À l'instar de la recherche d'information, le filtrage d'informations effectue une sélection des documents les plus similaires à la requête. Cependant et au contraire de la recherche d'information, la requête n'est pas nécessairement explicite, ni même liée à un geste ou un événement particulier. C'est par exemple le cas d'une recommandation spontanée du système de filtrage. La requête peut être représentée par un comportement, comme le fait de cliquer sur un item, ou être basée sur un profil d'intérêt qui peut aussi se baser sur le comportement de l'utilisateur ou sur d'autres informations. Le filtrage commence donc après la définition du besoin de l'utilisateur, il permet d'éliminer les documents qui peuvent ne pas intéresser l'utilisateur. Le filtrage offre à l'utilisateur un gain d'effort et de temps.

Le filtrage peut utiliser un calcul de similarité entre les documents ou entre les profils des utilisateurs. Cette distinction est à la base des trois principales approches de filtrage utilisées : le filtrage basé-contenu, le filtrage collaboratif et le filtrage hybride. Ces approches sont décrites dans ce qui suit.

1.4.1 Le filtrage basé sur le contenu

Un système qui utilise le filtrage basé contenu exploite seulement les représentations des documents et les informations qui peuvent être dérivées de ces documents. Un tel type de filtrage pourrait par exemple utiliser la similarité des documents dans une matrice termes-documents pour déterminer la pertinence d'un document. Si un utilisateur exprime un intérêt pour un document, les documents similaires seront jugés potentiellement pertinents aussi. C'est d'ailleurs la technique que nous allons adopter dans notre propre système, FCRC.

Pour mieux comprendre cette approche, nous allons étudier l'exemple du système PRES (acronym for Personalized Recommender System) [Maarten et Someren, 2000] qui est un système de recommandation basé-contenu. PRES crée des hyperliens dynamiques pour un site

web qui contient une collection de conseils pour l'amélioration de la personnalité. Le but de ces hyperliens dynamiques est d'aider les utilisateurs à trouver des items pertinents.

La figure 1.4 présente une structure simple du site web où le contenu des pages représente les feuilles de l'arbre tandis que les pages de navigation se trouvent en haut. PRES affiche ses recommandations par la création dynamique des liens hypertextes vers des contenus de pages qui contiennent les items pertinents.

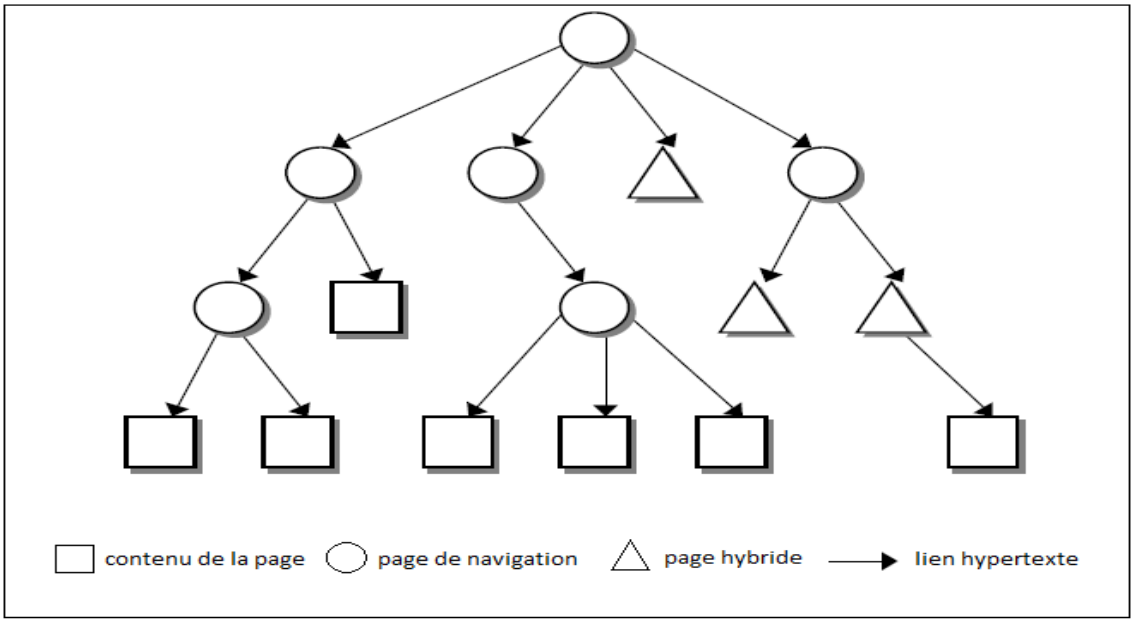


Figure 1.4 : Structure en arbre d'un simple site web. PRES [Maarten et Someren, 2000]

Un site Web peut être structuré en divisant ses pages web en des pages de contenu et des pages de navigation (voir figure 1.4). Une page de contenu fournit à l'utilisateur les items pertinents tandis qu'une page de navigation aide l'utilisateur à les rechercher. Les pages peuvent également être hybrides, elles fournissent à la fois le contenu ainsi que des fonctionnalités de navigation. Une page de navigation pour un utilisateur peut être une page de contenu pour un autre et visa versa.

PRES traite les données et classifie les items dans une classe positive nommée C (items pertinents). Par la suite il utilise les corrélations entre les contenus des items et les préférences des utilisateurs pour déterminer le degré de similarité.

PRES affiche ses recommandations dynamiquement par la création des liens hypertextes vers des pages de contenu qui contiennent les items pertinents. Il fait des recommandations en comparant chaque profil d'utilisateur avec le contenu de chaque document de la collection. La similarité entre le vecteur de profil et un document est déterminée par l'utilisation de la mesure du cosinus dans un espace vectoriel <terme, contenu du site web>.

Quelques limites des systèmes de filtrage basé contenu sont présentées par Berrut et Denos dans [Berrut et Denos, 2003]. La première difficulté pour ce type des systèmes c'est qu'il ne traite que des documents textes, nous ne pouvons pas indexer des documents multi media. De plus les systèmes basés contenu ne considèrent que le facteur thématique comme critère de pertinence malgré qu'il existe d'autres facteurs tels que : la fiabilité de la source d'information, la qualité scientifique et le degré de précision des faits présentés. Les auteurs évoquent aussi l'effet « entonnoir » qui restreint le champ de vision de l'utilisateur: à force de ne recommander que des items reliés à une seule et même thématique, et du fait que l'intérêt de l'utilisateur est déterminé à partir des items choisis, on créé un effet d'auto-entraînement qui restreint toujours plus la définition des intérêts de l'utilisateur et empêche l'exploration de thématiques différentes.

1.4.2 Le filtrage collaboratif

Le filtrage collaboratif est un parmi les technologies les plus populaires dans le domaine des systèmes de recommandation [Herlocker et al. ,2000]. Le filtrage collaboratif se base sur l'idée que les personnes à la recherche d'information devraient se servir de ce que d'autres ont déjà trouvé et évalué. Dans la vie quotidienne, si quelqu'un a besoin d'une information, il essaye de s'informer généralement auprès de ses amis, ses collègues, qui vont à leurs tours lui recommander des articles, des films, des livres, etc. Cette collaboration entre les gens permet d'améliorer l'échange de connaissances, mais cela prend beaucoup de temps vu que cette ressource d'information ne peut pas toujours à notre disposition. C'est à partir d'ici que l'idée de filtrage collaboratif été née, le besoin être d'automatiser et de rendre l'échange des expériences et des avis personnels de certains personnes utilisables par d'autres. Selon Golberg [Golberg, 2001] le filtrage collaboratif est l'automatisation des processus sociaux.

À la base du filtrage collaboratif, on utilise les choix explicites ou explicites des utilisateurs pour des items. Donc plutôt que d'utiliser une matrice termes-documents comme la recherche d'information le fait et comme l'approche basée-contenu, le filtrage collaboratif utilise la matrice des votes utilisateurs-items, où un vote peut être soit explicite (ex. fournir une cote à un film), soit explicite (ex. acheter un DVD du film).

Le tableau illustre un exemple de matrice-utilisateurs-items. Les valeurs peuvent représenter des votes ou des comportements. Dans cet exemple, l'item I3 n'a pas de vote pour l'utilisateur U1. On peut tenter de l'estimer soit avec une approche item-item ou une approche utilisateur-utilisateur comme expliqué dans les sections qui suivent.

Tableau 1.2 Matrice Items-utilisateurs

Utilisateur	Item			
	I1	I2	I3	I4
U1	5	1	**?**	2
U2	4	1	0	3
U3	4	2	1	2
U4	1	4	3	2

Pour mieux définir ce type de filtrage on se réfère aux travaux de Breese, Heckerman et Kadie dans [Breese, Heckerman et Kadie, 1998]. Ils décrivent plusieurs algorithmes conçus pour cette tâche, comprenaient des techniques basées sur les coefficients de corrélation, le calcul de similarité basée sur les vecteurs et méthodes bayésiennes statistiques. Le filtrage collaboratif peut prendre plusieurs formes : Item-Item et Utilisateur-Utilisateur.

1.4.2.1 Filtrage item-item (Linden et al, 2003)

Le filtrage item-item calcule la similarité des items dans la matrice utilisateur-items. Afin de déterminer les items les plus similaires à un item donné, cet algorithme construit un tableau des items similaires en cherchant les items populaires (que les clients ont tendance à les acheter). Une

matrice des produits entre les Items est construite, ces produits représentent le degré de similarité entre les items (paire par paire). On peut trouver des paires qui n'ont aucun client en commun. Dans le but de conserver le temps d'exécution et l'utilisation de la mémoire, il sera beaucoup mieux si on calcule le produit d'un item par rapport aux autres items au lieu de calculer les produits de tous les items à la fois :

La similarité entre les items peut être calculée de différentes méthodes, mais la méthode la plus commune est la mesure de cosinus des items (voir section 1.3.1.2).

1.4.2.2 Filtrage utilisateur-utilisateur

Les filtres collaboratifs utilisent des bases de données sur les préférences des utilisateurs. Généralement, un utilisateur est caractérisé par ses goûts, ses préférences et ses choix. A travers ses traces (des achats, des consultations de pages, des votes…). L'algorithme utilisateur-utilisateur cherche la similarité entre les profils des utilisateurs et recommande des items à un nouvel utilisateur. Ces recommandations sont basées sur le groupe le plus près et le plus représentatif d'un individu. Le principe de cette approche est de chercher la similarité entre les profils utilisateurs en se basant sur leurs préférences et leurs appréciations.

Le tableau 1.2 représente les valeurs des votes donnés par des quatre utilisateurs : U1, U2, U3 et U4 et leurs votes pour quatre items. L'utilisateur U1 n'a pas encore voté pour l'item I3 L'algorithme Utilisateur-Utilisateur permet d'estimer le vote de cet utilisateur U1 à l'item I3 en fonction des votes des autres utilisateurs. La valeur estimée E de l'utilisateur U1 pour un item I3, est la somme pondérée des votes des autres utilisateurs, Vi, qui ont des votes communs. Avec i est l'index de chaque vote.

Ainsi, pour estimer le vote à l'item I3, on utilise la formule suivante :

$$v_{ij} = \ moy(v_i) + \ k \sum_{i}^{m} w_i v_{ij}$$

Où :

v_{ij} : vote de l'utilisateur i à l'item j

$moy(v_i)$: vote moyen de l'utilisateur i

k : constante pour normaliser la somme des w_i à 1 (dans le cas où le poids est un cosinus, par exemple, ce sera la somme des cosinus).

w_i : poids de l'utilisateur i. Ce poids peut être le cosinus entre le vecteur ligne de l'utilisateur pour lequel on désire estimer le vote et l'utilisateur i. Ce peut aussi être la corrélation ou une autre mesure.

m : nombre de votes.

1.4.3 Le filtrage hybride

Le filtrage hybride combine les deux types de filtrage basé sur le contenu et le filtrage collaboratif.

1.5 Algorithmes et techniques pour la recommandation de cours

Avant tout, il nous semble opportun de présenter les différents modèles employés pour les systèmes de recommandation de cours qui va nous permettre de faire advenir l'originalité de notre approche. Chaque système de recommandation de cours utilise des techniques et des algorithmes différents. Nous révisions ces systèmes en détails dans les pages à venir.

1.5.1 La théorie des réponses aux items (IRT)

Le système PEL-IRT utilise la théorie des réponses aux items (IRT) dont le but d'adapter le choix de cours recommandé au niveau d'expertise. IRT établit que la probabilité de succès à un item (une question ou un exercice) est une fonction d'une seule habileté désignée par thêta θ [Birnbaum, A., 1968]. La relation entre l'habileté et la probabilité de succès est décrite par la courbe caractéristique de l'item (voir figure 1.5) On trouve en abscisse de la courbe, les valeurs des traits latents de l'individu (son niveau d'aptitude) désigné par thêta et en ordonné la probabilité que cet individu réponde avec succès à l'item.

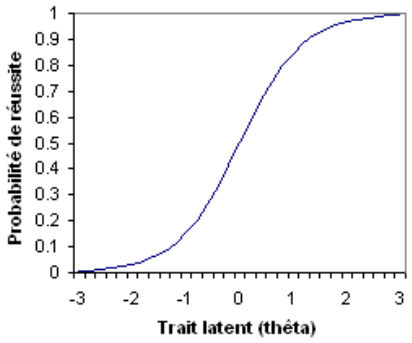


Figure 1.5 Courbe caractéristique de l'item

Cette courbe est modélisée par la fonction sigmoïde :

$$P(X|\Theta) = \frac{1}{1 + e^{-a(\theta - b)}}$$

Les paramètres *a* et *b* de cette fonction sont :

- *b :* le paramètre de difficulté.
- *a :* le paramètre de discrimination.

L'estimation de l'habileté d'un individu est faite à partir des succès et échecs à une série de questions. Par exemple, supposons trois questions : X_1, X_2 et X_3, dont les deux premières sont réussies et les deux secondes échouées, alors la probabilité d'une valeur donnée de θ est donnée par :

$$P(\theta|X_1 = 1, X_2 = 1, X_3 = 0) = P(X_1 = 1|\theta)\, P(X_2 = 1|\theta) P(X_3 = 0|\theta)$$

L'estimation du paramètre θ est faite en maximisant la probabilité ci-dessous. C'est ce que l'on nomme la technique de la maximisation de la vraisemblance.

Le modèle Rash [Birnbaum, A. (1968] est une version du modèle ci-dessus où le paramètre de discrimination, *a*, est omis.

1.5.2 Le raisonnement basé-cas [Aamodt et Plaza, 1994]

Le système AACORN applique une approche de raisonnement basé-cas afin afin d'effectuer des recommandations aux nouveaux étudiants. Nous le verrons à la section 2.1.3.

Le raisonnement basé-cas ou CBR (Case-based reasoning) est un paradigme de résolution des problèmes. CBR permet l'utilisation des connaissances spécifiques des anciennes expériences et des situations problématiques concrète (cas). Un nouveau problème est résolu par la recherche d'un cas similaire et le réutiliser en l'adaptant à ce nouveau problème. Un autre point très important, c'est que CBR est une approche incrémentale, soutenue par l'apprentissage. Chaque problème résolu sera sauvegardé et disponible pour être réutilisé dans d'autres cas.

Les méthodes de raisonnement basé-cas fonctionnent comme suit :

1. Identifier la situation du problème actuel.
2. Trouver un cas similaire.
3. Utiliser ce cas pour proposer une solution au problème actuel.
4. Évaluer la solution proposée, et mettre à jour le système par ce qui été appris suite à cette expérience.

Le paradigme CBR couvre une gamme de méthodes différentes pour l'organisation, la récupération, l'utilisation et l'indexation des connaissances. Le cycle CBR se décompose en quatre étapes (figure 1.6):

1. RÉCUPERER les cas les plus similaires.

2. RÉUTILISER les informations et les connaissances dans ce cas, pour résoudre le problème.

3. RÉVISER la solution proposée.

4. CONSERVER les parties de cette expérience risque d'être utile pour résoudre les problèmes futurs.

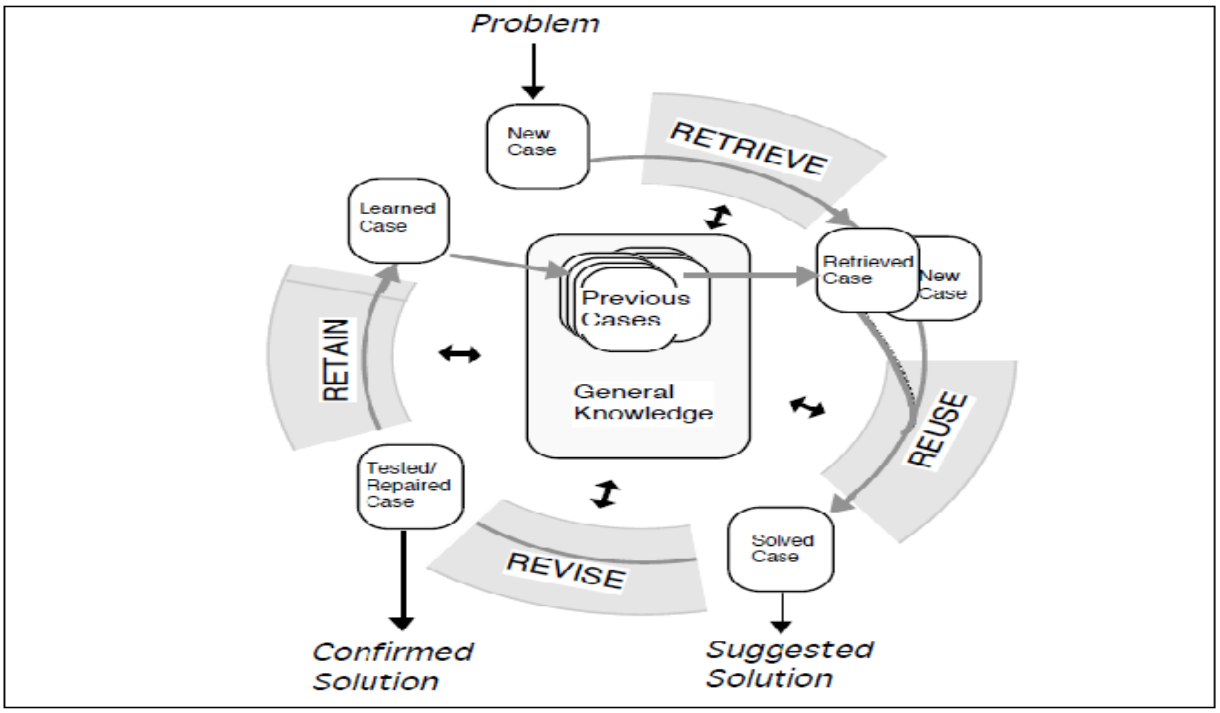


Figure 1.6 cycle de la méthode CBR [Aamodt et Plaza, 1994]

1.5.3 La recherche des règles d'association :

Le système de recommandation de cours Rare utilise des algorithmes pour définir des règles d'association entre les étudiants et les choix de cours. Ces règles sont obtenues par l'analyse de données. La recherche des règles d'association est une méthode très populaire dans le domaine de data mining. Elle permet de découvrir des relations entre les différentes variables dans de très importantes bases de données. Une règle d'association est défini sous la forme suivante :

Si **condition** alors **résultat**

Prenant un p exemple **:** Si les cours C1 et C2 apparaissent simultanément dans un plan d'études d'un étudiant alors le cours C3 apparaît.

Pour associer des règles, différent algorithmes peuvent être utilisés tels que l'algorithme Eclat (basé sur l'intersection des ensembles), l'algorithme OPUS et l'algorithme Apriori. Généralement, le data mining est un processus qui utilise une variété d'outils d'analyse de

données pour découvrir des patterns et des relations entre les données, qui peuvent à leur tour être utilisés pour faire des prédictions [Edelstein, 2003].

CHAPITRE 2 RECOMMANDATION ET FILTRES D'INFORMATION

Le chapitre précédent décrit les fondements algorithmiques des systèmes de recommandation. Le présent chapitre porte sur les systèmes qui effectuent des recommandations de cours. Il est organisé comme suit : dans la section 2.1 nous allons présenter les différents systèmes de recommandation de cours. Ensuite nous allons présenter un récapitulatif résumant les avantages et les inconvénients de chaque système. Finalement, nous expliquons notre choix d'adopter une approche basée-contenu.

2.1 Les systèmes de recommandation de cours

Notre objectif est de développer un système de recommandation de cours pour les étudiants. Il existe plusieurs systèmes similaires. Nous présentons dans ce qui suit six systèmes de recommandation.

2.1.1 PEL-IRT [Chen, Lee, & Chen, 2005]

La personnalisation de l'apprentissage est un objectif très important pour les systèmes tutoriels, spécialement pour les apprentissages basés sur le web. En général la plupart des systèmes personnalisés prennent en considération le niveau de connaissances, les intérêts et l'historique des choix pour fournir des services personnalisés. Dans [Chen, Lee, & Chen, 2005] une approche collaborative PEL-IRT effectue l'analyse des paramètres de difficultés des supports du cours afin d'adapter le contenu présenté au niveau de la connaissance de l'utilisateur. PEL-IRT est un système personnalisé d'apprentissage basé sur la théorie des réponses aux items qui analyse à la fois les difficultés des supports de cours et les capacités des apprenants dans le but de fournir un apprentissage personnalisé.

La fonction caractéristique de l'item utilisée pour PEL-IRT est celle proposée par Rasch (voir section 1.5.1). Ce modèle fait appel à un seul paramètre : le paramètre de difficulté. Encore, l'estimation de vraisemblance maximale (MLE) est appliquée pour estimer les niveaux de compétences des capacités des apprenants basées sur leurs feedbacks explicites (section 1.5.1). L'architecture du système PEL-IRT contient deux parties principales (voir figure 2.1) :

1) La partie en arrière qui analyse les capacités des apprenants (CA) et qui sélectionne les supports de cours à partir de la base de données de cours en se basant sur l'estimation de la CA.

2) La partie en avant : cette partie représente l'interface agent qui identifie les statuts des apprenants, transfère les requêtes et retourne la liste de cours recommandés aux étudiants.

PEL-LRT fournit une interface de recherche afin d'aider les étudiants à trouver des supports de cours dans une unité de cours spécifiée. Les apprenants pourraient aussi utiliser la recherche par mots clés.

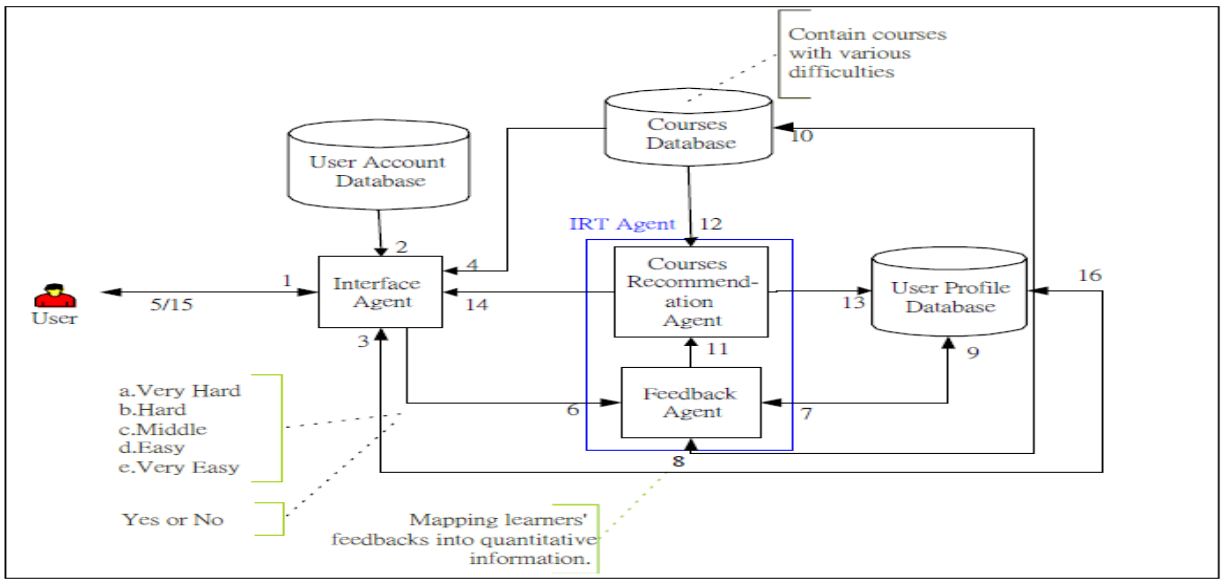


Figure 2.1 Architecture du système PEL-IRT [Chen, Lee, & Chen, 2005]

Pour évaluer PEL-IRT, un prototype a été publié en ligne. Ce prototype fournit des services personnalisés et facilite l'évaluation des recommandations. Les résultats expérimentaux prouvent d'une part que la théorie des réponses appliquée peut réaliser l'apprentissage personnalisé selon les cours visités par les étudiants et leurs réponses sur les deux questionnaires, et d'autre part que PEL-IRT a fourni des recommandations personnalisées basées sur l'analyse des capacités des apprenants en améliorant l'efficacité chez eux.

Le contenu d'un cours portant sur l'algorithme Perceptron est utilisé pour évaluer les résultats expérimentaux. L'évaluation contient 20 documents de cours, 18 apprenants inscrits au programme de mastère et qui sont connectés à PEL-LRT. Parmi eux on trouve 13 apprenants

qui ont déjà pris le cours du réseau de neurones. L'étude de Chen, Lee et Chen [Chen, Lee, & Chen, 2005] comporte une évaluation des compréhensions des apprenants. Le degré moyen de la compréhension des supports de cours proposé par PEL-LRT était de 0.8. Ce résultat indique que le niveau de compréhension des apprenants était élevé. L'évaluation par les étudiants des difficultés moyennes des supports du cours sont respectivement : 1.764 et 1.596 ce qui montre que la plupart des apprenants pensent que les supports du cours recommandés appartiennent à des difficultés modérées. Les auteurs concluent que PEL-IRT a adapté avec succès le niveau des supports de cours en fonction du CA des apprenants.

2.1.2 RARE [Bendakir, & Aimeur 2006]

Le système RARE décrit dans [Bendakir & Aimeur, 2006] est un système de recommandation de cours pour les étudiants de cycles supérieurs du département informatique de l'Université de Montréal. Il se concentre sur l'efficacité de l'incorporation de techniques de *data mining* (forage de données) pour la recommandation de cours.

RARE utilise des données réelles concernant les cours et les étudiants du département informatique. Il analyse les comportements des étudiants suivant leurs choix d'études en formalisant des règles associés implicitement (section 1.5.4). Ensuite il propose des recommandations de cours jugés pertinents et il demande aux étudiants de les évaluer.

L'architecture de RARE est composée de deux phases (voir tableau 2.1) :

Tableau 2.1 Les phases de RARE

La phase en ligne :	Processus de data mining	Extraire les règles des données qui décrivent les sélections des anciens étudiants.
La phase hors ligne :	Interaction de RARE avec l'utilisateur.	Utiliser les règles extraites pour inférer de nouvelles recommandations de cours.

Les données :

Les informations concernant les étudiants et les cours sélectionnées sont filtrées pour être organisées d'une façon plus succincte aux utilisateurs. L'interaction entre les utilisateurs et le

système se fait à chaque utilisation :

RARE demande à la première utilisation de l'étudiant de s'enregistrer et de fournir des informations initiales. Après la première utilisation l'étudiant doit vérifier les cours qu'il a suivis récemment et évaluer les cours suivis. RARE intègre certains critères supplémentaires tels que l'évaluation des cours, la fidélité (dont il se sert pour évaluer les recommandations) de l'étudiant et le poids des cours.

Principe de l'algorithme :

La première étape consiste à effectuer la classification des données, puis à effectuer la recherche des règles d'association (section 1.5.3). La classification permet de construire un arbre de décision (ou règles de classification) qui classifie chaque élève. Puisque les données recueillies sont relatives aux cours suivis, il est possible de construire les profils des étudiants qui représentent les différents chemins des cours choisis. Chaque chemin commence à partir de la racine, passe par les nœuds représentant les cours et conduit à une feuille marquée avec un laboratoire de recherche. Le processus de recommandation est effectué en suivant les chemins dans l'arbre selon les cours suivis par un étudiant. Si un seul cours a été suivi, RARE cherche des chemins dans les deux sens qui commencent par ce cours. Les autres cours de ces chemins ainsi que les laboratoires seront recommandés. Si deux cours ont été suivis par un étudiant, RARE cherche des chemins qui commencent par ces deux cours. Ensuite, il recommande le troisième et le quatrième cours, ainsi que les laboratoires [Bendakir & Aimeur, 2006].

L'algorithme C4.5 [Quinlan, 1993] est la technique de utilisée pour construire l'arbre de décision. Cet algorithme est disponible dans le logiciel WEKA (Waikato Environnement Knowledge analysis) [Witten & Frank, 2005]. WEKA est un package open source pour l'apprentissage automatique et le *data mining* développé à l'Université de Waikato en Nouvelle-Zélande.

Évaluation expérimentale:

L'évaluation repose sur une comparaison entre les cours proposés par RARE et les cours réellement choisis par l'étudiant selon les données colligées. Un cours est jugé pertinent s'il a été choisi par l'étudiant dans les données et l'évaluation consiste à mesurer la couverture et la précision des recommandations par rapport aux cours jugés pertinents. Les dossiers de 40 étudiants ont été échantillonnés aléatoirement. Ces étudiants ont été inscrits au programme de

maîtrise informatique à l'Université de Montréal entre l'an 2000 et 2006.

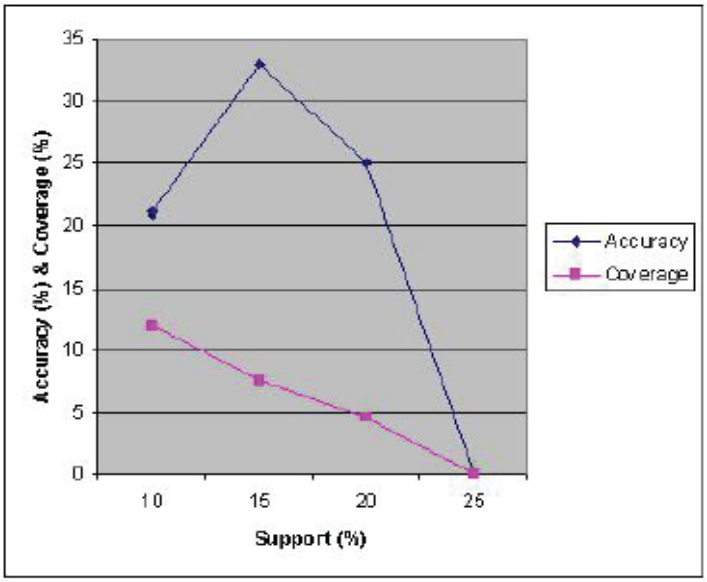


Figure 2.2 Précision et couverture de RARE [Bendakir & Aimeur, 2006] (RARE donne deux cours suivis par les étudiants).

Dans la figure 2.2, la précision commence à diminuer lorsque la valeur de soutien augmente au-dessus de 15%. Les étudiants ont seulement deux cours en commun dans le meilleur des cas. Les règles de la forme A et B → C sont beaucoup moins importantes et rares. Par exemple, A et B se réfèrent généralement à des cours qui sont communs au sein d'une spécialité (ou d'un laboratoire de recherche). Ils sont très instructifs, tandis que C est un cours lié au domaine d'application de l'étudiant et porte donc moins d'informations. Nous pouvons voir que d'un seuil de soutien entre 10% et 15% donne des résultats qui permettent d'atteindre l'objectif. Ainsi, pour recommander des cours, les règles générées dans cet intervalle sont conservées dans la base de règles. Si cette approche est appliquée à un autre domaine, la fixation de seuils de soutien et de confiance dépend de l'application et les données traitées.

2.1.3 AACORN [Sandvig & Burke 2006]

Tout comme RARE, AACORN (Academic Advisor COurse Recommendation eNgine) est un système fondé sur l'analyse des historiques des cours afin de les recommander aux nouveaux étudiants. Il se base sur la réutilisation des expériences des anciens étudiants trouvés

dans les historiques des relevés de notes pour adapter une solution pour chaque nouvelle recommandation. AACORN fournit des propositions raisonnables avec un domaine de connaissance limité. Il utilise la distance d'édition (section 1.3.6) entre deux historiques de cours, ainsi que d'autres méthodes pour calculer les similarités entre les historiques des cours.

L'approche proposée dans [Sandvig & Burke 2006] vise à permettre aux étudiants d'atteindre leurs objectifs académiques et de satisfaire les exigences des cours pour l'obtention de leurs diplômes. Elle utilise les données de choix de cours pour effectuer des recommandations qui respectent les programmes existants et offre ainsi une méthode qui permet de tenir compte automatiquement de changements aux programmes et facilite du coup sa maintenance.

Le système AACORN se base sur l'idée que les étudiants similaires dans leur intérêt ou dans leur cheminement académique choisissent les mêmes cours. Deux étudiants sont inscrits au même programme ayant des intérêts similaires auront besoin des mêmes cours. C'est une méthode qui utilise des connaissances spécifiques des expériences précédentes afin d'adapter des anciens plans aux nouvelles situations avec une petite modification c'est le raisonnement basé-cas (CBR) décrit dans la section 1.5.2.

AACORN est capable de traiter les informations liées aux étudiants, aux programmes académiques et aux historiques des cours. Les évaluations initiales du système montrent que les résultats sont prometteurs malgré que cela nécessite une étude plus approfondie. . Le système inclut les fonctionnalités essentielles pour un système de recommandation, par contre, à plus long terme, le système nécessitera plusieurs améliorations majeures. En particulier, AACORN devra inclure une contrainte pour le filtrage des recommandations des cours lorsque l'étudiant ne satisfait pas aux conditions préalables. Il recentrera également le problème de satisfaction des préférences en permettant à l'étudiant d'interagir avec le système.

2.1.4 CourseRank [Parameswaran, Venetis, & Garcia-Molina, 2010]

CourseRank est un système de recommandation de cours de l'université de Stanford décrit dans [Parameswaran, Venetis, & Garcia-Molina, 2010], il évalue et planifie les programmes académiques. Les cous recommandés par CourseRank doivent satisfaire les contraintes et les exigences des cours.

CourseRank recommande les cours populaires et les cours déjà pris par des étudiants

similaires. Ce travail est spécifiquement pour les exigences des cours. Selon les auteurs [Parameswaran, Venetis, & Garcia-Molina 2010], développer un système de recommandation nécessite de prendre en considération certaines contraintes. L'algorithme max-flow est utilisé pour la vérification de ces contraintes. Pour chaque étudiant un graphe G sera créé : on considère un nœud S comme une source et un nœud O comme objectif. Deux ensembles A et B représentent respectivement : les cours et les exigences. Prenons l'exemple suivant (voir figure 2.3) du modèle de base des exigences du cours. Pour obtenir leur diplôme avec une majeure en CS les étudiants doivent répondre à plusieurs sous-exigences. Les auteurs se limitent à un seul sous-ensemble des exigences : R1, R2 et R3. Où R1 représente les sous-exigences théoriques, R2 représente les sous-exigences liées aux bases de données et R3 pour les sous- exigences du système.

Supposons également que l'étudiant peut suivre des cours de l'ensemble {a, p, d, i, o}.

Où

-a est un cours d'introduction dans les algorithmes,

-p est un cours système de principes de bases de données.

-d représente un cours d'introduction aux bases de données.

- i représente un cours de mise en œuvre d'un système de bases de données.

- o est un cours de système d'exploitation.

Ensuite, les sous exigences pourraient être comme suit:

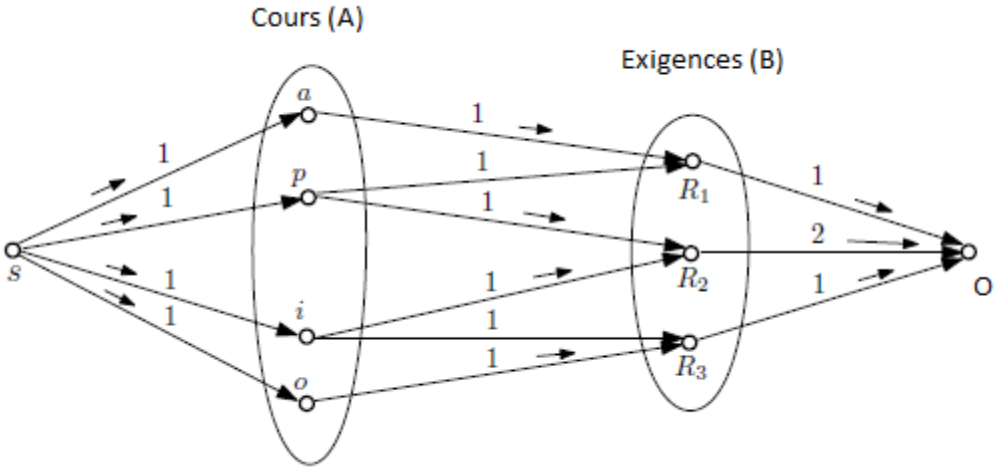


Figure 2.3 Les flux du modèle de base G pour un étudiant E[Parameswaran, Venetis, & Garcia-Molina, 2010].

Dans la figure 2.3, les chiffres 1 et 2 indiquent le nombre de cours maximales. Les flèches au-dessus des bords indiquent les flux de sortie. Par exemple, l'étudiant E doit prendre un des cours a et p si on veut respecter la sous exigence R1.

R1: prend 1 de {a, p}

R2: prend 2 à partir de {p, i}

R3: prend 1 de {i, o}

CourseRank utilise aussi l'algorithme ILP (Inductive Logic Programming) pour vérifier les exigences et recommande les cours les plus pertinents. Il utilise la fonction de score pour le calcul de la similarité entre un ensemble de cours au lieu de calculer le score entre chaque cours individuellement. Le minimum des cours requis par un étudiant sera identifié à l'aide du graphe G.

Les auteurs ont évalué les deux techniques approximatives fondées sur les flux et la technique ILP exacte pour la vérification des exigences et pour les recommandations. Cette évaluation était basée sur un ensemble de données sur les relevés de notes pour les cinq étudiants majeurs. Ils ont constaté que les deux techniques permettent de recommander les cours que les étudiants finissent par prendre. La technique ILP performe légèrement mieux (et plus précis), mais nécessite un temps d'exécution plus long.

2.1.5 Course recommender system SCR [Ekdahl, Lindstrom, & Svensson 2002]

[Ekdahl, Lindstrom, & Svensson 2002] présentent le système SCR qui utilise les méthodes d'apprentissage machine pour la création d'un système de recommandation de cours très fiable et qui apprend à partir des choix des cours faits par des étudiants. L'objectif de cette étude est d'estimer les intérêts des étudiants et de créer un système qui peut être étendu avec des nouvelles fonctionnalités.

Le système SCR (Course Recommender) est une implémentation standard de l'architecture client/serveur (voir figure 2.4). Le client interagit avec une interface graphique

contenant la logique et une partie pour la manipulation des requêtes des clients pour les recommandations. SCR implémente un modèle de réseau bayésien (voir section 1.3.4).Il recommande les cours qui ont des probabilités élevées d'être pertinent. Le réseau bayésien représente un modèle utilisateur où les nœuds sont des cours. À mesure que des cours sont suivis par l'étudiant, le modèle est mis à jour afin de calculer la probabilité de pertinence des autres cours.

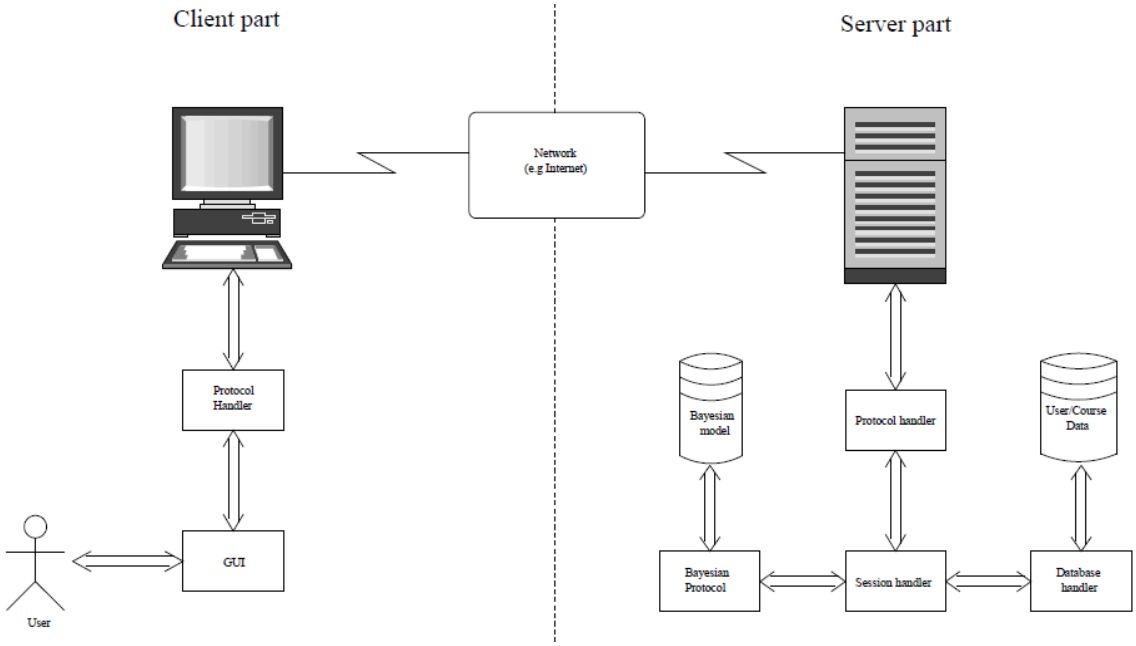


Figure 2.4 Aperçu de l'architecture du système [Ekdahl, Lindstrom, & Svensson 2002]

SCR utilise des informations qui peuvent être recueillies explicitement ou implicitement.

Les techniques explicites exigent que l'utilisateur remplisse un formulaire avec certaines informations. Les techniques implicites recueillent les informations par les observations du comportement de l'utilisateur. Les données recueillies sont ensuite transmises au modèle bayésien afin d'inférer les cours pertinents et non pertinents.

Les auteurs affirment que leurs expériences avec le modèle développé pour les recommandations des cours démontrent que cette méthode peut être utilisée avec succès pour la recommandation de cours.

2.2 Récapitulatif

Dans ce qui suit, nous présentons un tableau récapitulatif sur les avantages et les inconvénients de chaque système.

Tableau 2.2 Avantages et inconvénients des systèmes de recommandation de cours

Système de recommandation	Avantages	Inconvénients
SCR	• Prend en considération des critères tels que la disponibilité du cours, les cours indésirables.	• Exige une période d'entraînement. • Le problème de démarrage à froid. • Demande beaucoup de ressources.
AACORN	• Fournit des recommandations basées sur les historiques des cours	• Exige une période d'entrainement. • Le problème de démarrage à froid. • Demande beaucoup de ressources.
PEL-IRT	• La recherche des matériels d'un cours dans une unité spécifique.	• Demande beaucoup de ressources. • Inscription requise.
RARE	• Demande moins de ressource que les autres systèmes. • Découvre les relations cachées entre les données.	• Ne prends pas en considération les contraintes et les exigences des cours. • Nécessite un grand volume de données.
CourseRank	• Prend en considération les exigences des cours. • Peut être utilisé dans d'autres domaines.	• Complexité de calcul et d'exécution. • Algorithme complexe.

Vu que nous nous intéressons à la recommandation de cours dans notre étude, on peut se référer à quelques systèmes de recommandation de cours qui ont utilisé le filtrage collaboratif dont le but de ressortir notre contribution. RARE (section 2.1.2) se base sur l'analyse des comportements des anciens étudiants. Il joue le rôle de l'intermédiaire entre les étudiants qui ont déjà complété leurs études et les nouveaux étudiants qui ne savent pas beaucoup de choses sur les cours offertes. RARE transfert les expériences de certains étudiants pour que d'autres y bénéficient. Pareille pour les autres systèmes : AACORN, SCR et PEL-IRT. Ils emploient le filtrage collaboratif, mais ils utilisent différentes mesure de similarité entre les profils des étudiants.

Une grande partie des étudiants à Montréal sont des étudiants étrangers. Ils viennent de différents pays avec des différentes connaissances et formations. Par conséquence, leurs besoins sont différents même s'ils ont des intérêts similaires. Recommander des cours en se basant sur les anciennes expériences des étudiants ne prend pas en considération cette diversité, à moins qu'on ajoute de nouveaux critères de similarité tels que : l'école, la formation, le pays etc. Le système RARE par exemple, est basé en quelque sorte sur les évaluations de ses utilisateurs, or que les étudiants peuvent ne pas répondre aux questionnaires, ils peuvent aussi évaluer les recommandations arbitrairement. Ce qui cause un problème pour certaines sources d'information utilisées par RARE.

Encore, considérer les exigences et les contraintes (le cas du système CourseRank) dans le processus de recommandation de cours n'est pas un facteur important. L'étudiant au cycle supérieur peut consulter le système une seule fois, lors de sa première session. Il peut planifier son plan d'études pour toute la durée des études (il ne va pas prendre tous les cours dans une seule session donc ça ne va être pas un problème si un cours n'est pas disponible pour la première session). La plupart des cours offerts aux étudiants de 1er cycle sont disponibles pendant au moins deux sessions. Réellement, les cours pertinents existent déjà mais le problème est comment trouver les meilleures offres et où les trouver ? C'est sur cette idée qu'on doit se concentrer. Si on arrive à trouver les sigles ou les titres des cours pertinents, les autres informations liées aux cours telles que la description, l'horaire et la disponibilité vont être accédées facilement.

À partir d'un objectif très précis, nous pourrions prévoir les besoins des utilisateurs et former une requête (un nom d'un produit, un identifiant, des mots clés…etc.). Prenant l'exemple

d’un recruteur à la recherche d’un nouvel employé. Chaque employé peut être représenté par son *c*urriculum vitae qui contient ses expériences et ses qualifications. Le filtrage qu’on doit employer pour ce cas est un filtrage basé sur le contenu *et* les contenus *sont les c*urriculum vitae des candidats. Pareille pour la recherche des cours, la meilleure façon de représenter un cours est par sa description. Nous devons traiter les besoins des étudiants d’une façon indépendante. Donc nous optons le filtrage basé sur le contenu pour notre modèle FCRC.

CHAPITRE 3 LE MODÈLE DE RECOMMANDATION FCRC

Dans ce chapitre nous présentons notre modèle FCRC et en détaillons les différentes étapes de la réalisation. Nous commençons par la présentation des outils de recherche des cours utilisés actuellement des universités : Polytechnique (Poly), Université de Montréal (UdeM), Université de Québec à Montréal (UQAM) et l'école de gestion HEC Montréal. Ensuite nous expliquons :

- La préparation des données : les descriptions des cours.
- La formation du corpus de validation.
- La procédure de génération des recommandations.

FCRC recommande les cours en se basant sur leurs descriptifs. Il nous faut donc extraire ces descriptions à partir des sites web des quatre universités situées à Montréal : UdeM, HEC, UQAM, Poly. Une fois les descriptions collectées, nous devons lemmatiser tous les termes pour ne garder que la lemme du mot. Par exemple, les mots comme « arpenteur », « arpentage » et « arpenter » seront transformés par « arpent ». Ce processus a pour effet de créer des similarités entre les mots qui autrement ne seraient pas reliés.

Ensuite, une matrice des termes lemmatisés par les descriptions de cours est crée. Cette matrice constitue un espace vectoriel termes-documents.

Nous allons utiliser le modèle d'espace vectoriel décrit dans la section 1.3.1 pour le calcul de similarité. Différentes mesures de similarité dans l'espace vectoriel seront utilisées pour calculer la similarité des descriptions de cours : cosinus, Dice et produit scalaire.

Notre approche est donc basée-contenu (voir section 1.4.1) et le contenu ici représente les descriptions des cours. Le principe est simple, si nous avons deux descriptions d1 et d2 similaires nous considérons que d1 peut être recommandé pour quelqu'un qui s'intéresse à d1, et vice-versa. L'approche brièvement décrite ci-dessus est détaillée dans le reste de cette section.

3.1 Les interfaces de recherche des cours actuels :

Dans ce qui suit, nous présentons les différentes interfaces de recherche des cours en ligne. Un script Python (Annexe 1) a été développé pour interroger ces interfaces et extraire les descriptions de cours. Ce script a été développé par des étudiants du cours INF8007 du département de génie informatique et il a été adapté à nos besoins. On peut considérer ce script comme *web crawler* : un robot qui suit les liens de pages web à la recherche de descriptions de cours.

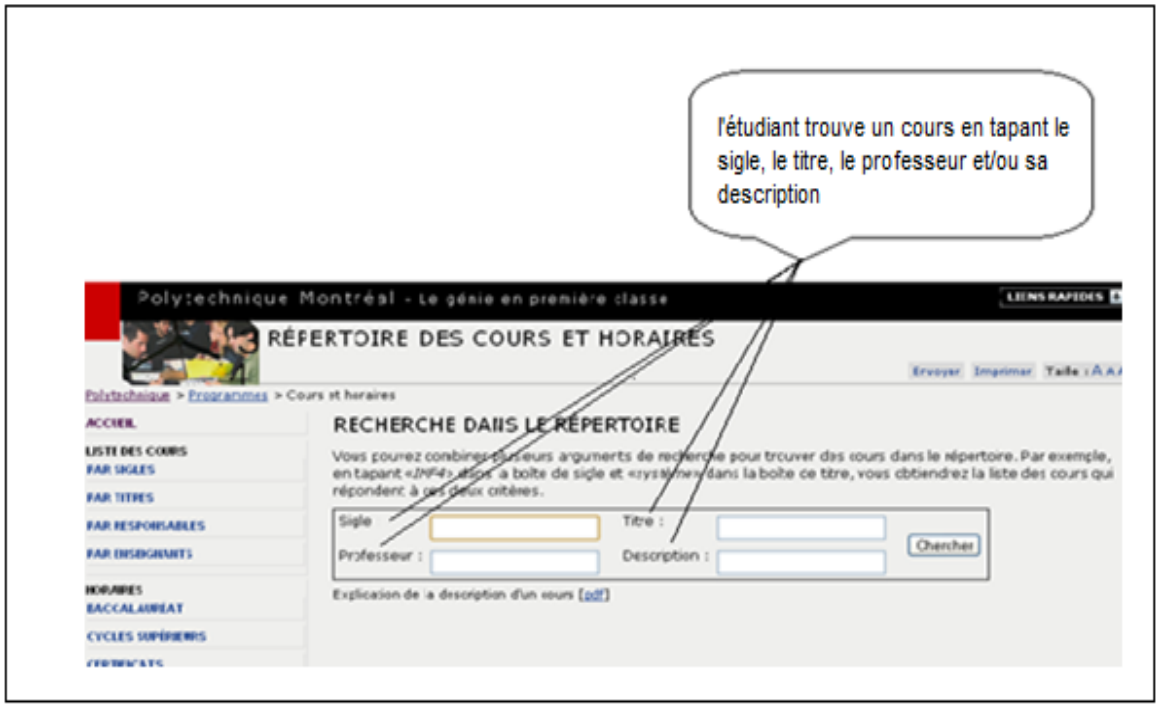


Figure 3.1 Interface de la recherche des cours de l'École Polytechnique.

La figure 3.1 illustre l'interface de recherche de cours de l'École Polytechnique. Cette interface permet d'effectuer des recherches par nom de professeur du cours, par département ou par sigle. En interrogeant cette interface par tous les sigles possibles (INF, LOG, MEC, etc.), il est alors possible de soutirer l'ensemble des descriptions de cours. Une technique similaire est aussi possible pour soutirer les cours des autres universités (voir les figures 3.2 à 3.4).

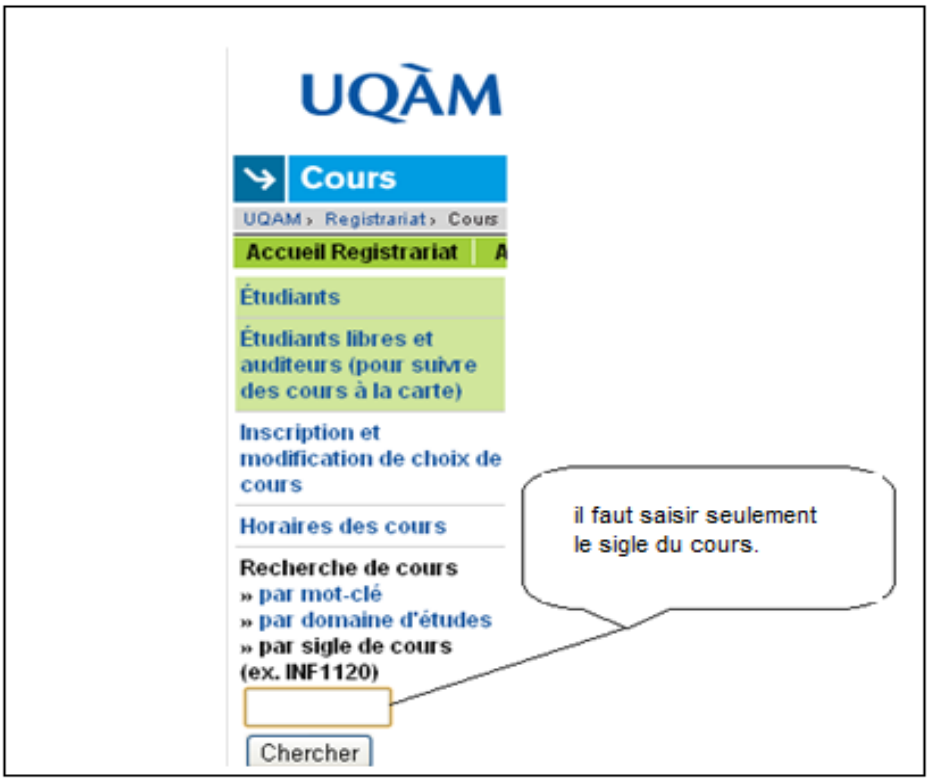


Figure 3.2 Interface de la recherche des cours de l'UQAM

Figure 3.3 Interface de la recherche des cours de l'Université de Montréal

Figure 3.4 Interface de la recherche des cours du HEC.

3.1.1 Collecte des descriptions des cours

Le script python extrait les descriptions des cours à partir des sites web suivants :

-Site web de l'école polytechnique de Montréal : http://www.polymtl.ca/

- Site web de l'Université de Montréal : http://www.umontreal.ca/

- Site web de l'université de Québec à Montréal : http://www.uqam.ca/

- Site web de l'HEC Montréal : http://www.hec.ca/

Pour le site de Polytechnique, le script Python va extraire tous ce qui se trouve entre les balises : <TITLE> ici c'est écrit le titre du cours</TITLE> et les balises : <DESCRIPTION> ici c'est écrit la description du cours</DESCRIPTION>. Pour les autres sites, il faut adapter le script afin d'extraire les bonnes balises qui varient d'un site à l'autre.

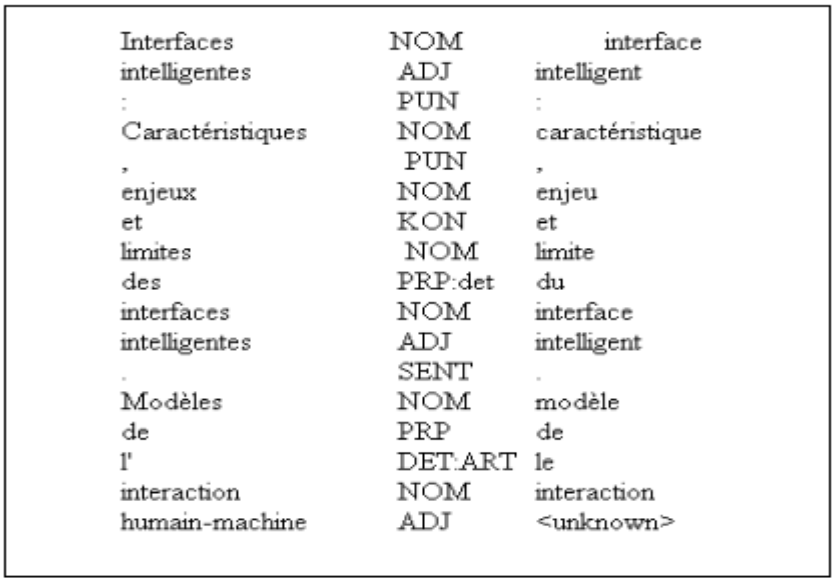
3.1.2 Lemmatisation

Nous avons déjà mentionné que les descriptions de cours sont lemmatisées. Précisons maintenant quelques détails techniques de cette lemmatisation.

Afin de capturer la similarité sémantique des mots, il faut les transformer en un lemme commun. Ainsi, nous devons regrouper et unifier et la représentation des mots de la même famille (nom, pluriel, verbe a l'infinitif…) par la lemmatisation. Différents outils existent à cette fin :Tree tagger ou Mallet (annexe 1). Nous choisissons l'outil Tree tagger (voir annexe 1) car il possède une syntaxe simple, facile à apprendre et il gère bien la mémoire dynamique. Nous allons présenter un exemple de lemmatisation pour un extrait de la description du cours INF6304.

La description avant la lemmatisation contient ces termes : Interfaces intelligentes : Caractéristiques, enjeux et limites des interfaces intelligentes. Modèles de l'interaction humain-machine.

Après la lemmatisation on trouve que certains termes ont été unifiés tel que le terme « intelligentes ». Cet adjectif n'est plus au pluriel, pareil pour les termes : limites et interfaces.



Interfaces	NOM	interface
intelligentes	ADJ	intelligent
:	PUN	:
Caractéristiques	NOM	caractéristique
,	PUN	,
enjeux	NOM	enjeu
et	KON	et
limites	NOM	limite
des	PRP:det	du
interfaces	NOM	interface
intelligentes	ADJ	intelligent
.	SENT	.
Modèles	NOM	modèle
de	PRP	de
l'	DET:ART	le
interaction	NOM	interaction
humain-machine	ADJ	<unknown>

Figure 3.5 Termes groupés et unifiés dans une seule représentation.

3.2 Création de la matrice termes-documents et calcul du TFIDF

La procédure de génération des recommandations de FCRC seront basées sur le calcul de similarité entre les cours. Ce calcul nécessite premièrement la construction d'une matrice termes-documents contenant les fréquences brutes, puis d'une matrice TFIDF où les fréquences sont transformées en multipliant la fréquence par le poids (TFIDF).

3.2.1 Matrice termes-documents

Les descriptions sont représentées par un ensemble de termes lemmatisés. On créé ensuite une matrice termes-document dans laquelle chaque colonne correspond à un terme unique et chaque ligne représente une description de cours. Chaque cellule contient la fréquence d'apparition du terme dans la description.

La matrice doit contenir les fréquences d'apparition FA de chaque terme dans chaque document. Si nous avons par exemple le terme « commande » qui apparait dans la description du cours c1 et c2 mais pas dans la description du cours c3, on écrit 1 dans les cellules qui associent les cours c1, c2 avec le terme « commande » et 0 dans la cellule qui associe c3 avec le terme « commande » (voir figure tableau 3.1).

Tableau 3.1 Extrait de la matrice Terme-Cours **M**

	T1	T2	T3	T4	T5	T6	T7	T8	T9
C1	1	1	1	1	0	0	1	0	0
C2	1	1	0	0	0	1	1	1	1
C3	1	0	0	0	1	0	1	0	1

Le tableau présente un extrait de la matrice termes-cours **M**. Les termes : terme 1, commande, terme 3 représentent les colonnes. Les cours C1, C2 et C3 représentent les lignes. Chaque cours devient donc un vecteur qui constitue les fréquences d'apparition de chaque terme de la matrice.

Quelques statistiques portant sur la matrice **M** sont rapportées.

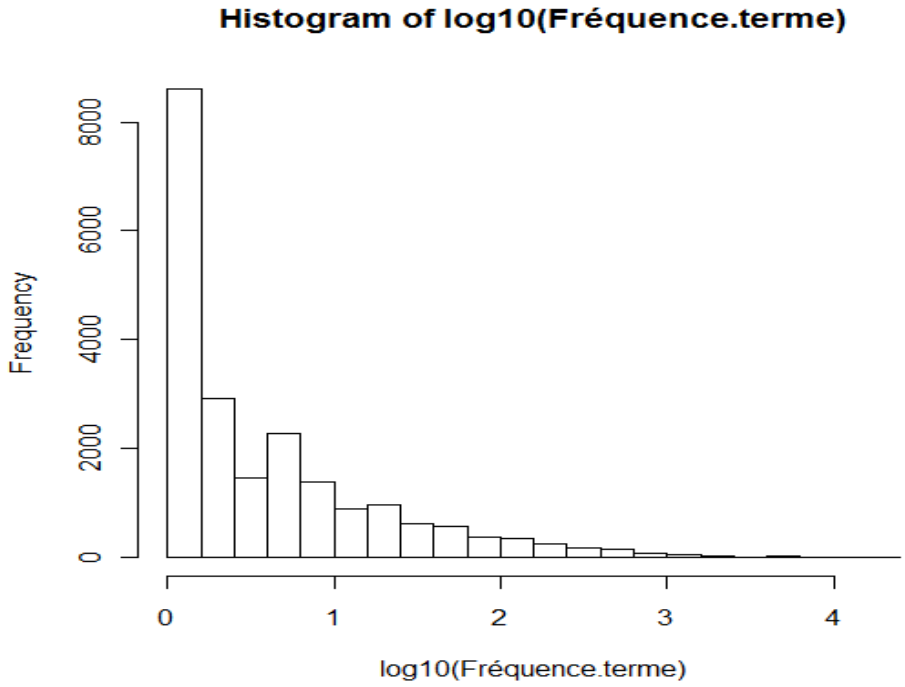


Figure 3.6 histogramme du log10 des fréquences de termes.

Tableau 3.2 Statistiques de la matrice **M**

Dimension de la matrice **M**	Moyenne	Écart type	Médiane
(16335, 21080)	43.12	917.55	2

La figure 3.6 présente un histogramme de la fréquence des termes lemmatisés et le tableau 3.2 fournit quelques statistiques sur la matrice.

3.2.2 Calcul du TFIDF

À partir de la matrice **M**, on définit une seconde matrice, **M.TFIDF**, qui contient les fréquences transformées par le poids des termes, c'est-à-dire le TFIDF décrit à la section 1.3.2. En définissant une matrice diagonale, **D**, de dimension $m \times m$, où m est le nombre de termes, et où le vecteur IDF des termes (c.-à-d. leur poids) est la diagonale de **D**, on peut alors définir la matrice **M.TFIDF** comme étant :

$$\boldsymbol{M.TFIDF} = (\boldsymbol{M}^T \boldsymbol{D})^T$$

Rappelons que la matrice **M** est de dimension $m \times n$ et que n est le nombre de documents.

3.2.3 Réduction de dimensions

Finalement, une troisième matrice, **M.SVD**, est aussi définie pour tenter d'extraire des dimensions latentes à la matrice **M.TFIDF** et ainsi d'obtenir de meilleurs résultats pour le calcul de la similarité de documents.

La technique d'indexation sémantique latente basée sur la décomposition en valeurs singulière (section 1.3.3) est utilisée à cette fin. La matrice **M.TFIDF** est décomposée en trois matrices :

$$\boldsymbol{M.TFIDF} = \boldsymbol{U\,\Lambda\,V^T}$$

Puis, une matrice $\mathbf{M.SVD}_d$ est alors obtenue en ne retenant que les d premières valeurs singulières de la matrice diagonale $\boldsymbol{\Lambda}$.

Trois valeurs de dimensions réduites ont été explorées : 20, 50 et 100.

3.3 Recommandation sur la base du calcul de la similarité avec un cours

À partir des matrices **M** et **M.TFIDF** et $\mathbf{M.SVD_d}$, une recommandation de cours peut-être effectuée sur la base de la similarité avec un cours donné. Deux mesures sont retenues à cette fin :

1. Le cosinus (section 1.3.1.2)
2. Le coefficient de Dice (section 1.3.1.3)

Nous explorerons les performances des différentes matrices et mesures de similarité dans l'expérimentation décrite et rapportée dans les prochains chapitres.

CHAPITRE 4 MÉTHODOLOGIE

Afin d'évaluer la capacité d'effectuer des recommandations pertinentes du modèle FCRC, nous effectuons une série d'expériences de simulations sous différentes conditions. Le principe général de ces expériences est d'évaluer dans quelle mesure le modèle FCRC peut retrouver les cours similaires à un cours donné dans le corpus de cours des universités montréalaises. Un corpus de validation composé des cours pertinents à quatre cours cibles est créé pour fin de comparaison avec les recommandations faites par le modèle FCRC.

4.1 Cours cibles

Quatre cours cibles sont identifiés pour évaluer le modèle. Il s'agit de :

1. Poly/IND6402 Interfaces humains-ordinateurs.

Description : Définition, classification et évolution des interfaces humain-ordinateur. Notions de compatibilité, accessibilité, sécurité, performance, utilisabilité, esthétique et expérience utilisateur. Méthodologie de conception centrée sur l'utilisateur. Normes, principes, critères et modèles de conception. Analyse des besoins et analyse contextuelle. Spécifications de l'utilisabilité. Modélisation, maquettage et prototypage. Styles d'interaction humain-ordinateur. Dispositifs d'entrée de données et de pointage. Présentation d'informations. Fonctionnalités de soutien à l'utilisateur. Patrons de conception. Méthodes d'inspection ergonomique. Tests d'utilisabilité. Suivi de l'utilisabilité et de l'expérience utilisateur.

2. Poly/INF6304 Interfaces intelligentes.

Description : Caractéristiques, enjeux et limites des interfaces intelligentes. Modèles de l'interaction humain-machine et de l'utilisateur : connaissances, intérêts et préférences, buts et plans. Recherche d'information semi-structurée : filtres collaboratifs et sémantiques, agents de recherche. Aide et assistance : systèmes conseils, documents adaptatifs, dialogue coopératif et tuteurs intelligents. Interfaces sensibles au contexte. Validation des interfaces intelligentes.

3. Poly/INF6410 Ontologies et web sémantiques.

Description : Notions de base en logique propositionnelle et logique des prédicats. Logiques descriptives. Mécanismes d'inférence. Langages et modèles de données pour le web sémantique : langages de balisage et de transformation de documents électroniques, langage de description de ressources, langage de représentation d'ontologies. Ontologies standards. Méthodologie pour la construction d'une ontologie. Validation d'une ontologie. Applications du web sémantique : annotation et indexation sémantique de documents, outils de recherche.

4. Poly/IND6412 Ergonomie des sites web.

Description : Historique, développement et caractéristiques de l'Internet. Types de sites Web et caractéristiques de leurs interfaces. Langage HTML et technologies de mise en page et d'interactivité. Analyse des besoins préalable à la conception. Méthodologie de conception centrée sur l'utilisateur, normes et accessibilité. Prototypage d'interfaces. Architecture et navigation. Modes de dialogue. Conception des pages-écrans. Comportements et performance humaine liés aux sites Web. Utilisabilité des interfaces.

Le choix de ces cours est motivé par le fait que nous avons l’expertise pour identifier l’ensemble des cours pertinents à Poly comme à l’Université de Montréal et dans les autres universités, ce qui n’est pas le cas si l’on avait choisi des cours d’autres facultés ou même d’autres départements. Nous estimons que le fait qu’ils soient tous des cours d’informatique ne devrait pas fortement influencer la validité des résultats, il faut néanmoins souligner qu’il s’agit d’une limite à cette étude.

4.2 Cours pertinents

Pour chacun des cours cibles, un ensemble de cours similaires a été trouvé en cherchant :

- Les cours préalables.
- Les cours de la même spécialité.
- Les cours enseignés par un même professeur.

On a demandé aussi à quelques étudiants qui ont suivis au moins un des cours de L de nous proposer des cours similaires dans d’autres écoles. Un professeur connaissant bien le domaine des cours a aussi contribué à identifier les cours les plus pertinents.

Un total d'environ 40 cours pertinents a donc été pour l'ensemble des cours INF6304, INF6410, IND6402 et IND6412.

De plus, tous les cours retournés par le modèle ont été évalués pour leur pertinence afin de s'assurer que, même si un cours n'était pas identifié comme pertinent au préalable par la méthode ci-dessus, il était effectivement non pertinent. Nous avons choisi ces cours

4.3 Valeurs de pertinence

La pertinence d'un cours est basé sur la similarité avec un cours cible. Nous avons défini trois valeurs de similarité :

1. **0** : le cours n'est pas relié au cours cible.
2. **1** : le cours ne traite pas directement de la même matière que le cours cible mais les sujets sont connexes et pourraient s'avérer pertinents ou complémentaires.
3. **2** : le cours traite des mêmes sujets que le cours cible dans une proportion importante.

Les résultats de comparaison entre les recommandations de FCRC et les cours pertinents porteront donc à la fois sur le seuil de 0,5 et de 1.

4.4 Simulations et comparaisons

Comme mentionné au chapitre précédent, trois types de matrices sont crées et deux types de mesures de similarité sont utilisées. Nous obtenons donc 6 conditions expérimentales pour lesquelles nous rapportons deux mesures :

Tableau 4.1 Conditions expérimentales.

	Similarité	
	Coefficient de Dice	Cosinus
M (fréquences brutes)		
M.TFIDF (fréquences avec poids IDF)		
M.SVD (valeurs singulières)		

Pour chaque condition expérimentale, nous rapportons la valeur de l'aire sous la courbe ROC, nommée AUC décrite ci-dessous.

4.5 Mesures de performance

Afin d'avoir le système de recommandation le plus efficace possible, il faut sélectionner la mesure qui permet d'avoir des recommandations relatives aux besoins des utilisateurs. Nous possédons ainsi une méthode pour l'évaluation de chacune de nos mesures de similarité. La méthode se base sur le calcul de l'aire sous la courbe ROC et la mesure Accuracy. Nous allons utiliser le package ROCR qui est un outil flexible permettant de créer des courbes de performances 2D en combinant deux mesures. La courbe ROC est peut être généré grâce au Package ROCR. Un système de recommandation de cours est évalué en fonction de la pertinence des documents retrouvés.

4.5.1 La courbe ROC

La courbe ROC et une mesure classique dans le domaine de la classification. Elle représente l'évolution de la sensibilité (taux de vrais positifs) en fonction de la spécificité (taux de faux positifs) quand on fait varier le seuil de classification [Fawcett, 2006]. C'est une courbe croissante entre le point (0,0) et le point (1, 1) et en principe au-dessus de la première bissectrice.

Si l'on attribue au hasard des valeurs de pertinence, cette courbe prendra l'allure d'une diagonale de pente 1. Au contraire, une performante parfaite amène la courbe du point (0,0) au point (0,1) où elle demeure jusqu'au point (1,1).

4.5.2 L'aire sous la courbe ROC (AUC Area Under the Curve)

AUC est la mesure de la surface située sous le tracé de la courbe ROC. C'est un indicateur de la qualité de la prédiction (1 pour une prédiction idéale, 0.5 pour une prédiction aléatoire). Cette mesure a l'avantage de ne pas avoir à fixer de seuil pour décider de la classification (pertinente ou non pertinente) pour une valeur de cosinus, de coefficient de Dice ou pour toute autre mesure de similarité.

4.6 Logiciel de calcul et de simulation

Tous les calculs de similarité ainsi que les évaluations sont faits avec le logiciel R-2.13.0 (voir annexe 1) qui est à la fois un langage et un environnement de travail permettant de réaliser des calculs et des analyses statistiques.

CHAPITRE 5 RÉSULTATS ET ÉVALUATION

Nous présentons dans cette section, les résultats de la simulation des recommandations pour les quatre cours cibles et pour les différentes conditions expérimentales décrites dans le tableau 4.1.

5.1 Cosinus avec les matrices M et M.TFIDF

Nous commençons par la mesure de cosinus (voir 1.3.1.2) sur la matrice des valeurs brutes **M** puis sur la matrice **M.TFIDF**. On cherche à trouver les cours similaires aux cours suivants : interfaces intelligentes (INF6304), ontologies et web sémantiques (INF6410), interface humains-ordinateurs (IND6402) et le cours ergonomie des sites web (IND6412). Dans ce qui suit, nous rapportons les 10 cours les plus similaires à ces cours selon le cosinus et en utilisant les matrices des valeurs brutes **M** et la matrice **M.TFIDF**.

Tableau 5.1 Liste des cours similaires au INF6304 (interfaces intelligentes) trouvés par le calcul des cosinus sur **M**.

ECOLE/SIGLE	TITRE DU COURS	Cosinus	Pertinence
o UQAM/INF4150	o Interfaces personnes machines.	0,640	2
o UQAM/MGL830	o Ergonomie des interfaces usagers.	0,610	2
o UQAM/EDM9161	o Interaction humain-ordinateur.	0,584	2
o Poly/IND6409	o Interfaces humains-ordinateurs spécialisées	0,582	2
o Poly/IND6402	o Interfaces humains-ordinateurs.	0,576	2
o Poly/LOG2420	o Anal. et conc. des interfaces utilisateurs	0,571	2
o UQAM/ORH3000	o Méthodes de Rech. App. A la GRH.	0,557	0
o Poly/IND6412	o Ergonomie des sites web.	0,551	2
o HEC/6-090-08	o Atelier de recherche en contrôle de gestion.	0,550	0
o UQAM/INF7530	o Interfaces personne-machine.	0,548	2

D'après le tableau 5.1, le 7éme et le 9éme cours recommandés ne sont pas pertinents. Ces cours ne sont pas similaires au cours INF6304. Les cours du reste sont pertinents. Après cette vérification manuelle on peut dire que la précision est 8/10.

Tableau 5.2 Liste des cours similaires au INF6410 (ontologies et web sémantique) trouvés par le calcul des cosinus sur **M.**

ECOLE/SIGLE	TITRE DU COURS	Cosinus	Pertinence
o UQAM/SYS841	o Systèmes experts	0,739	2
o UQAM/DES1530	o Projet de design I	0,730	0
o UQAM/ORH3000	o Méthodes de Rech. App. A la GRH.	0,728	0
o Poly/IND3903	o Projet intégrateur : systèmes d'information	0,723	0
o UQAM/JUR2508	o Systèmes et doc. Juri. canadiens	0,721	0
o UQAM/GHR6700	o Gestion de l'hébergement	0,721	0
o UdM/LNG2240	o Pragmatique	0,719	0
o UQAM/MSL9001	o Méthodologie en recherche en muséologie	0,717	0
o Poly/SB100	o Méthodologie de résolution de problèmes.	0,717	0
o UQAM/ENV7140	o Principes de gestion intégrée des ressources.	0,717	0

Le tableau 5.2 présente la liste des cours similaires au cours ontologies et web sémantiques, la plus part des cours ne sont pas pertinents. Seuls les cours SYS841 est similaire à INF6410, la précision est de 1/10.

Les tableaux 5.3 et 5.4 contiennent les résultats pour les deux autres cours, IND6402 et IND6412.

Tableau 5.3 Liste des cours similaires au IND6402 (Interfaces humains-oridnateurs) trouvés par le calcul des cosinus sur **M.**

ECOLE/SIGLE	TITRE DU COURS	Cosinus	Pertinence
o UQAM/AVM5900	o Didactique de l'enseign. des arts plas. aux adultes	0,766	0
o UQAM/DES7520	o Compos. humains, anth. et activ. des utilisateurs	0,758	0
o Poly/IND3903	o Projet intégrateur : systèmes d'information	0,758	1
o UQAM/EDM9161	o Interaction humain-ordinateur.	0,755	2
o UQAM/INF4150	o Interfaces personnes machines.	0,753	2
o Poly/IND8844	o Ergonomie avancée	0,753	2
o Poly/SB100	o Méthodologie de résolution de problèmes.	0,751	0
o UQAM/STA9850	o Concept de sys. en sci. de la Ter. de l'atmosphère	0,750	0
o UQAM/DES7530	o Normes et sécurité dans les transports	0,750	0
o UQAM/PSY9433	o Herméneutique et psychothérapie	0,750	0

Tableau 5.4 Liste des cours similaires au IND6412 (ergonomie des sites web) trouvés par le calcul des cosinus sur **M.**

ECOLE/SIGLE	TITRE DU COURS	Cosinus	Pertinence
o HEC/4-720-00	o Conception de sites Web	0,783	2
o UQAM/INF4150	o Interfaces personnes machines.	0,770	2
o UQAM/AVM5900	o Didactique de l'enseign. des arts plas. aux adultes	0,767	0
o Poly/LOG4420	o Conception de sites web dynam. et transact.	0,751	2
o Poly/IND8844	o Ergonomie avancée	0,746	2
o Poly/IND6402	o Interfaces humains-ordinateurs.	0,745	2
o Poly/GCH6311	o Concept. gest. des centres de trait.des sols	0,744	0
o UQAM/PSY9433	o Herméneutique et psychothérapie	0,744	0
o Poly/IND3303	o Projet intégrateur : systèmes d'information	0,736	0
o UQAM/INF7350	o Ingénierie des logiciels	0,735	1

Tableau 5.5 Liste des cours similaires au cours INF6304 pour **M.TFIDF**

ECOLE/SIGLE	TITRE DU COURS	Cosinus	Pertinence
o UQAM/INF4150	o Interfaces personnes machines.	0,398	2
o Poly/IND6409	o Interfaces humains-machines spécialisées	0,385	2
o UQAM/MGL830	o Ergonomie des interfaces usagers	0,338	2
o UdM/IFT2905	o Interfaces personne-machine.	0,324	2
o UQAM/INF7530	o Interfaces personnes machines.	0,307	2
o Poly/LOG2420	o Anal. et conc. des interfaces utilisateurs.	0,282	2
o UQAM/INF7510	o Système à base de connaissance.	0,260	2
o Poly/IND6402	o Interfaces humains-machines spécialisées	0,236	2
o Poly/IND6412	o Ergonomie des sites web.	0,227	2
o UdM/CHM3404	o Surfaces interfaces colloïdes	0,197	0

Les tableaux 5.4 à 5.8 contiennent les résultats, mais cette fois pour les calculs de cosinus avec la matrice **M.TFIDF**. Les résultats sont nettement meilleurs.

Le cours CHM3404 n'est pas similaire à INF6304, il est classé 10éme dans la liste des recommandations. Le résultat avec la matrice tf.idf est meilleur que celui avec la matrice **M** avec une précision de 9/10.

Les recommandations obtenues en travaillant avec la matrice des poids **M.TFIDF** sont meilleures que ceux avec la matrice des valeurs brutes **M**.

Tableau 5.6 Liste des cours similaires au cours INF6410 pour **M.TFIDF**

ECOLE/SIGLE	TITRE DU COURS	Cosinus	Pertinence
o UdM/LNG2080	o Sémantique du français	0,363	2
o UQAM/LIN8203	o Sémantique 1	0,358	2
o UQAM/LIN3557	o Sémantique II	0,352	2
o UQAM/PHI4352	o Logique informelle	0,335	2
o UdM/LNG3050	o Sémantique : théorie et description	0,330	2
o UQAM/INF7870	o Fondement logiques de l'informatique	0,292	2
o UdM/ORA1534	o Langage 1	0,282	1
o UQAM/PHI3512	o Sémantique et pragmatique	0,276	1
o UdM/LNG1000	o Introduction aux langages formels	0,272	2
o UQAM/LIN1400	o A la découverte du langage	0,261	1

Tableau 5.7 Liste des cours similaires au cours IND6402 pour **M.TFIDF**

ECOLE/SIGLE	TITRE DU COURS	Cosinus	Pertinence
o Poly/IND6409	o Interfaces humains-ordinateurs spécialisées	0,402	2
o Poly/IND6412	o Ergonomie des sites web	0,392	2
o Poly/LOG2420	o Anal. Et conc. des interfaces utilisateurs	0,346	2
o UQAM/INF4150	o Interfaces personnes machines.	0,320	2
o UQAM/INF7530	o Interfaces personne-machine	0,299	2
o UQAM/MGL830	o Ergonomie des interfaces usagers	0,276	2
o UdM/COM2571	o Interfaces et scénarisation	0,245	2
o UQAM/PSY5890	o Utilisations de l'ordinateur en psychologie	0,240	1
o UQAM/INF7135	o Administration des informations et des données	0,239	0
o Poly/INF6304	o Interfaces intelligente.	0,236	2

Tableau 5.8 Liste des cours similaires au cours IND6412 pour **M.TFIDF**

ECOLE/SIGLE	TITRE DU COURS	Cosinus	Pertinence
o HEC/4-720-00	o Conception de sites Web	0,576	2
o Poly/LOG4420	o Conception de sites web dynam. et transact	0,452	2
o UdM/IFT1946	o Sites Web avancés avec Frontpage	0,442	2
o Poly/IND6409	o Interfaces humains-ordinateurs spécialisées.	0,430	2
o UQAM/INF4150	o Interfaces personnes machines.	0,405	2
o Poly/IND6402	o Interfaces humains-ordinateurs	0,392	2
o UdM/IFT1945	o Internet et création de pages Web	0,392	2
o UdM/INU3051	o Information et sites Web	0,324	2
o UQAM/MGL830	o Ergonomie des interfaces usagers	0,308	2
o UQAM/INF7530	o Interfaces personne-machine	0,292	2

Pour le cours IND6412, la précision des recommandations avec la matrice **M** était de 7/10, mais si on regarde le tableau 5.8, on remarque la différence entre ces deux listes de recommandations. La précision augmente avec la matrice **M.TFIDF** pour atteindre la valeur 1.

Les recommandations obtenues en travaillant avec la matrice des poids **M.TFIDF** sont meilleures que ceux avec la matrice des valeurs brutes **M**.

5.2 Coefficient de Dice (C.Dice) avec M.TFIDF

La deuxième mesure sera le calcul des coefficients de Dice. Cette mesure a donné de bons résultats, les résultats de M.TFIDF sont rapportés par souci de concision mais que les résultats globaux sont rapportés dans le tableau 5.13. Les cours similaires aux cours INF6304 et INF6410 sont presque les mêmes que ceux trouvés par le calcul des cosinus.

Tableau 5.9 Liste des cours similaires au cours INF6304 pour **M.TFIDF**

ECOLE/SIGLE	TITRE DU COURS	C.DICE	Pertinence
o UQAM/INF4150	o Interfaces personnes machines.	0,366	2
o UQAM/MGL830	o Ergonomie des interfaces usagers.	0,340	2
o UQAM/INF7510	o Systèmes à base de connaissance	0,339	2
o UQAM/INF7530	o Interfaces personne-machine.	0,251	2
o Poly/LOG2420	o Anal. Et conc. Des interfaces utilisateurs.	0,248	2
o Poly/IND6409	o Interfaces humains-ordinateurs spécialisées.	0,232	2
o UQAM/EDM9161	o Interaction humain-ordinateur.	0,197	2
o Poly/IND6402	o Interfaces humains-machines.	0,196	2
o UdM/IFT2905	o Interfaces personne-machine	0,195	2
o Poly/IND6412	o Ergonomie des sites web	0,188	2

On peut remarquer que les résultats obtenus pour les coefficients de Dice sont très proches des résultats des cosinus et même meilleure ayant une précision égale à 1.

Tableau 5.10 Liste des cours similaires au cours INF6410 pour **M.TFIDF**

ECOLE/SIGLE	TITRE DU COURS	C.DICE	Pertinence
o UQAM/LIN8203	o sémantique 1	0,288	2
o UQAM/SYS841	o systèmes experts.	0,272	2
o UQAM/LIN1400	o A la découverte du langage	0,264	2
o UQAM/LIN3557	o sémantique II.	0,250	2
o UQAM/PHI3512	o Sémantique et pragmatique	0,240	2
o UQAM/PHI1007	o Introduction à la logique.	0,234	2
o UQAM/PHI3508	o Logique intermédiaire	0,225	2
o UQAM/LIN9400	o sémantique 2	0,220	2
o UQAM/PHI4352	o logique informelle	0,218	2
o Poly/LOG4420	o Conception de sites web dynam. et transact.	0,202	2

De même pour le cours INF6410, les résultats obtenus sont très bons, le coefficient de Dice à donner des résultats meilleurs que ceux des cosinus.

Tableau 5.11 Liste des cours similaires au cours IND6402 pour **M.TFIDF**

ECOLE/SIGLE	TITRE DU COURS	C.DICE	Pertinence
o UQAM/PSY5890	o Utilisations de l'ordinateur en psychologie	0,423	1
o Poly/IND6412	o Ergonomie des sites web	0,411	2
o Poly/LOG2420	o Anal. Et conc. des interfaces utilisateurs	0,386	2
o UQAM/INF4150	o Interfaces personnes machines.	0,373	2
o HEC/30-760-96	o Ordinateurs et réseaux	0,356	0
o UQAM/MGL830	o Ergonomie des interfaces usagers	0,353	2
o UQAM/DES7520	o Composants humains, anthropométrie	0,337	2
o UQAM/INF7135	o Administration des infos. et des données	0,325	0
o UQAM/KIN6960	o Introduction a l'ergonomie	0,320	2
o UQAM/INF7530	o Interfaces personne-machine	0,310	2

Tableau 5.12 Liste des cours similaires au cours IND6412 pour **M.TFIDF**

ECOLE/SIGLE	TITRE DU COURS	C.DICE	Pertinence
o HEC/4-720-00	o Conception de sites Web	0,662	2
o Poly/LOG4420	o Conception de sites web dynam. Et transact	0,510	2
o UQAM/INF4150	o Interfaces personnes machines.	0,471	2
o Poly/IND6402	o Interfaces humains-ordinateurs.	0,411	2
o UQAM/MGL830	o Ergonomie des interfaces usagers	0,392	2
o UQAM/INF7212	o Introduction aux systèmes informatiques	0,370	0
o UQAM/AVM3301	o Arts médiat. Interac. Ubiquité et virtualité	0,347	2
o UdM/IFT1946	o Sites Web avancés avec Frontpage	0,333	2
o Poly/IND6409	o Interfaces humains-ordinateurs spécialisées	0,328	2
o UQAM/ETH1406	o Méthodo. Nouvelles tech. Et milieu scolaire	0,320	0

5.1 Évaluation d'ensemble avec ROC et AUC

Nous allons présenter dans ce qui suit la courbe ROC tracée pour différentes conditions et conformément au tableau 4.1. Cette fois, l'ensemble des 4 cours sont utilisés pour tracer la courbe ROC en utilisant les 10 premières recommandations pour chaque cours, tout comme dans la section précédente. Ces résultats sont rapportés dans les figures 5.1 à 5.6.

Chaque figure présente en abscisse les vrais positifs et en ordonné les faux positifs. Elle résume quatre courbes ROC : la première et la deuxième courbe exprime la performance en travaillant sur la matrice des valeurs **TFIDF** mais avec des seuils différentes qui sont respectivement :

S1 : 0,5 et

S2 : 0

Nous attribuons la valeur 1 si un cours est jugé très similaire, 0,5 si le cours et relativement similaire et on attribue un 0 si le cours n'est pas similaire. Les seuils S1 et S2 permettent respectivement d'identifier les cours moyennement pertinents et les cours pertinents.

La troisième et la quatrième courbe présentent la performance en travaillant sur la matrice des valeurs brutes. On commence par les courbes ROC générées à partir des résultats des cosinus avec la matrice TFIDF et la matrices des valeurs brutes. Nous prenons aussi les valeurs 0 et 0,5 pour les seuils S1 et S2.

Dans ce qui suit, on trace la courbe ROC des résultats obtenus pour les matrices **M** (qui contient tous les termes sans aucune modification.) et **M.TFIDF** avec les deux mesures de similarité : le cosinus et le coefficient de Dice.

1. La mesure de Cosinus**:**

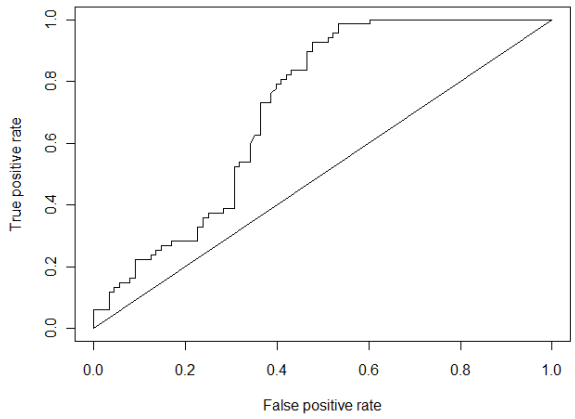


Figure 5.1 La courbe ROC pour la matrice **M** (S1).

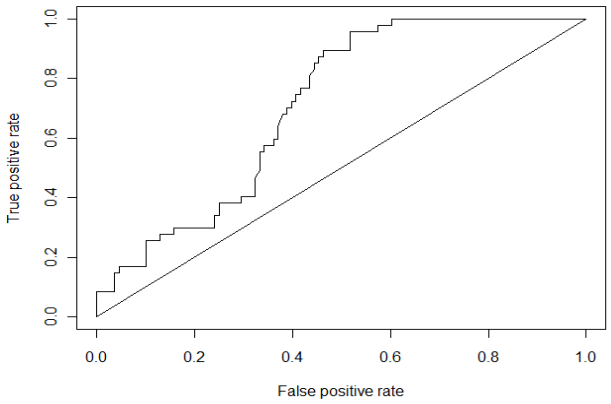


Figure 5.2 La courbe ROC pour la matrice **M** (S2).

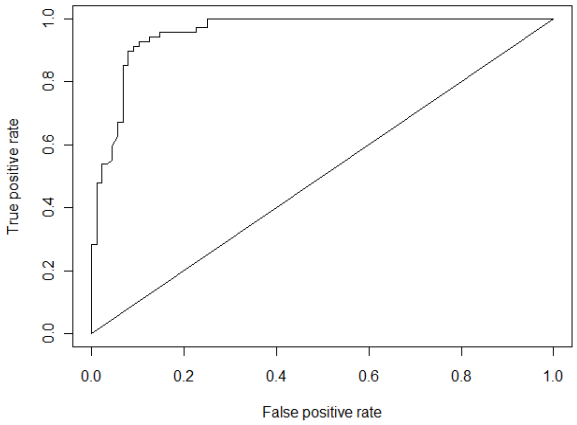


Figure 5.3 La courbe ROC pour la matrice **M.TFIDF** (S1).

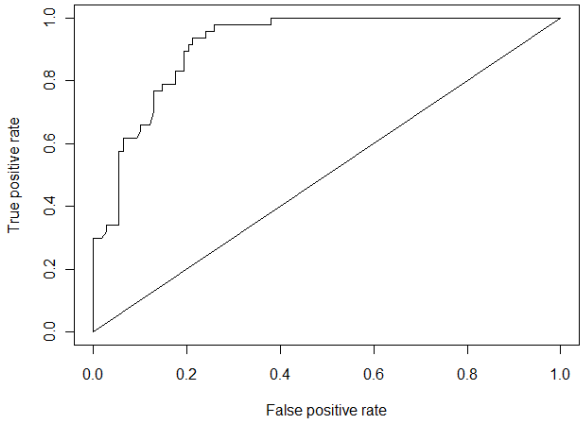


Figure 5.4 La courbe ROC pour la matrice **M.TFIDF** (S2).

A noter que les seuils S1 et S2 sont utilisés juste pour classifier les cours : des cours vrais positifs, des vrais négatifs, faux positifs et des faux négatifs.

Le coefficient de Dice

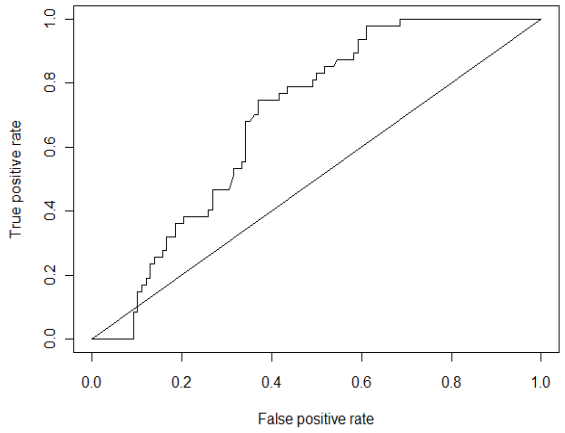


Figure 5.5 La courbe ROC pour la matrice **M**

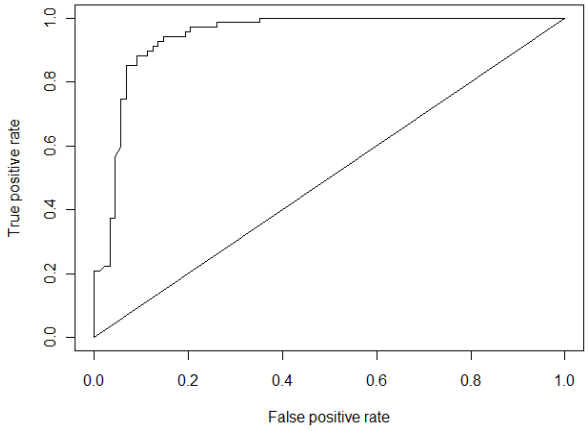


Figure 5.6 La courbe ROC pour la matrice **M.TFIDF**

Le calcul de l'AUC (aire sous la courbe ROC) fournit un indice de performance global. Le tableau 5.13 résume les moyennes des performances (pour les quatre cours) des deux mesures pour les matrices **M** et **M.TFIDF**.

Tableau 5.13 AUC pour **M** et **M.TFIDF** avec le cosinus et le coefficient de Dice

	M		**TFIDF**	
	Seuil		**Seuil**	
	S1= 0	S2= 0,5	S1=0	S2=0,5
Cosinus	0,71	0,70	0,95	0,91
C.Dice	-	0,69	-	0,94

Le tableau 5.13 nous permettent de noter plusieurs choses. Premièrement, les performances de toutes les mesures de similarité sur la matrice TFIDF sont meilleures que celles sur la matrice des valeurs brutes. Deuxièmement, les résultats obtenus suite aux calculs des cosinus et le calcul des coefficients de Dice sont très proches. Pour la mesure de coefficient de Dice on a calculé la performance avec un seuil de 0,5 car tous les coefficients de Dice sont supérieurs à 0. Ce

coefficient a performé mieux que le cosinus pour les quartes cours avec une AUC égale à 0,94 et 0,91 pour le cosinus.

5.2 Études exploratoires avec la méthode de la décomposition des valeurs singulières SVD

Comme nous l'avons vu à la section 2.2.2, la méthode de décomposition en valeurs singulières d'une matrice permet de réduire les dimensions et de projeter les valeurs originales par rapport à ces dimensions réduites. Cette technique revient en quelque sorte à calculer le niveau d'appartenance des termes par rapport aux dimensions latentes (thèmes) des documents. Nous explorons donc cette technique pour notre contexte et rapportons quelques résultats sommaires.

Cependant, comme la matrice **M** est très grande, nous rencontrons des problèmes de limites de mémoire avec la librairie R. Pour palier à ce problème, nous devons travailler avec un extrait de la matrice **M**. Le choix de l'extrait est aléatoire. On a pris 1000 cours auxquels on a ajouté les cours pertinents, de telle sorte qu'on aura une petite matrice de 1030 cours plutôt que les 16335 que nous avions originalement. Dans l'ensemble des cours pertinents on trouve des cours pertinents pour deux requêtes (un cours pertinent à la fois pour INF6304 et pour INF6410 ce qui explique l'ajout de 30 cours pertinents seulement).

Tableau 5.14 L'AUC de ROC pour la matrice M.SVD avec K= 20, K=50 et K=100

	M.SVD (sans aucune modification)			**M.TFIDF.SVD**		
	Dimension K			**Dimension K**		
	20	**50**	**100**	**20**	**50**	**100**
Cosinus	0,63	0,68	0,68	0,78	0,83	0,86
C.Dice	0,31	0,35	0,41	0,28	0,32	0,39

D'après le tableau 5.14, l'attribution des poids aux termes et l'utilisation d'une dimension K=100 ont permis d'augmenter la performance de 0,63 à 0,86. Cette technique demande plus d'effort et

des travaux afin d’apporter de l’amélioration, malgré que les résultats obtenus précédemment ont été presque parfaits.

Sous-section additionnelle.

CONCLUSION

Le domaine des interfaces de recommandation est un domaine très étendu et par conséquent difficile à cantonner. En premier lieu, nous avons essayé à travers ce mémoire de mettre la lumière sur les aspects les plus importants de ce domaine par la présentation:

- Des interfaces intelligentes.
- Des stratégies de recherche d'information : les mesures de l'espace vectoriel, l'inférence bayésienne.
- Des techniques de filtrage d'information telles que : le filtrage collaboratif, le filtrage hybride et le filtrage basé-contenu.
- On a aussi révisé les systèmes de recommandation de cours existants notamment, les techniques employées pour la génération des recommandations et le processus d'évaluation des recommandations fournis par ces systèmes.

Nous avons aussi présenté les différentes étapes de réalisation de notre modèle FCRC : la collecte des descriptions, la lemmatisation et la création de la matrice Termes-Documents.

On a vu aussi l'intérêt que l'on peut avoir à mettre en place un filtrage basé sur le contenu pour la recommandation de cours. Selon les évaluations des différentes méthodes de calcul de similarité entre les cours, la performance du cosinus est de 0.91si nous travaillons sur la matrice **M.TFIDF**. Encore, le calcul des coefficients de Dice a performé mieux que le calcul de cosinus avec une valeur de 0,94. À partir de la première évaluation, nous sommes arrivés à conclure que le cosinus et le coefficient de Dice sont deux mesures qui performent très bien au niveau des recommandations des cours fournis. Ce qui a permis d'avoir ces résultats appréciés est l'attribution des poids aux termes (TFIDF). Cette technique de pondération a augmenté la pertinence d'un terme en fonction de sa rareté au sein de l'ensemble des descriptions de cours et ceci a été confirmé par la remarquable croissance au niveau de la performance de 0,70 à 0,91.

Ainsi, notre modèle FCRC n'exige pas beaucoup de ressources d'informations. Il utilise un algorithme simple et efficace.

Comme perspectives de recherche, nous proposons d'intégrer le modèle FCRC avec l'un des systèmes de recommandation de cours qui exigent une grande quantité d'informations tels que :

RARE, AACORN. Ce modèle peut être considéré comme une solution au problème de démarrage à froid.

Le développement des systèmes de recommandation de cours nécessite plus d'efforts, plus d'évaluation des systèmes existants afin d'identifier et de gérer les déficiences que peuvent rencontrer ces systèmes. L'objectif principal de ces systèmes est de fournir le nombre maximal des recommandations pertinentes.

RÉFÉRENCES

Aamodt, A. et Plaza,E. (1994, Novembre). Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *AI Communications*, 7(1): 39-59.

Parameswaran, A., Venetis, p. et Garcia-Molina, H. (2011). Recommendation Systems with Complex Constraints: A CourseRank Perspective. *Transactions on Information Systems.* Volume 29(4).

Waern, A. (1997).What is an Intelligent Interface.

Bendakir, N., Aïmeur, E. (2006) Using Association Rules for Course Recommendation. *AAAI Workshop on Educational Datamining.* Boston.

Berry, M. J. & Linoff, G. (1997). Data Mining. Techniques appliquées au marketing, à la vente et aux services clients. *Informatiques.* Paris: Masson/InterEditions.

Breese, J.; Heckerman, D.; and Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. *In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, 43–52.

Catherine Berrut et Nathalie Denos. (2003). *Filtrage collaboratif, de l'aide intelligente à la Recherche d'Informations, Hermes-Lavoisier, chapitre 8*, pp30.

Chen, C. M.; Lee, H. M.; and Chen, Y. H. (2005). Personalizede-learning system using item response theory. *Computers and Education* 44(3):237–255.

Traduction de Daniel Arapu.Han, Kamber et Pei, Jiawei Han, Micheline Kamber, Jian Pei. (2011). *Data Mining Concepts and Techniques (3rd ed).* Morgan Kaufmann.

Denning, P. J. ‘Electronic Junk’. (1982). *Communications of the ACM 25(3),* 163–16.

Edelstein, H. (2003) Data mining in depth: Description is not prediction. *DM Review*.

Ekdahl, M.; Lindstr¨om, S.; and Svensson, C.(2002). A student course recommender. Master of Science Programme, Lulea University of Technology, Department of Computer Science and Electrical Eng neering / Division of Computer Science and Networking.

Fawcett, T. (2006.). An introduction to ROC analysis. *Pattern Recognition Letters* 27, 861–874.

Fuhr, Norbert. Probabilistic models in information retrieval. (1992). *The computer Journal*. 35(3):243-255.

Salton, G., Yang, c. et Wong, A. (1975). Un modèle vectoriel pour l'indexation automatique. *Communications of the ACM*. 18 (11) :613-620.

Grossman, D. A., et Frieder, O. (2004). *Information retrieval (2ème édition)*. Springer.

Harman, D. (1992a). *Ranking Algorithms*, chapter 14. Prentice-Hall, Englewood Cliffs, New Jersey.

Harman, D. (1992b). User-friendly systems instead of user-friendly front-ends. *Journal of the American Society for Information Science*, 43:164–174.

Herlocker, J. , Konstan, J. et Riedl, J. (2000, December). Explaining collaborative filtering recommendations, *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, p.241-250, Philadelphia, Pennsylvania, United States.

Breuker, J. (1990). EUROHELP: Developing Intelligent Help Systems. *Report on the P280 ESPRIT Project EUROHELP*. Lawrence erlbaum associates, publishers.

Goldberg, K., Roeder, T., Gupta, D. et Perkins, C. (2001, July). Eigentaste: A Constant Time Collaborative Filtering Algorithm. *Information Retrieval*. 4(2), 133-151.

Landauer et al. (1998) Introduction to LSA. *Discourse Processes*. 25, p. 259-284.

Landauer, T. K. et Dumais, S. T. (1997). A solution to Plato's problem: The Latent Semanctic Analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review*. 104 , 211-140.

Linden, G., Smith, B., et York, J. (2003). Amazon.com recommendations: item-to-item collaborative filtering. *Internet Computing, IEEE*. vol. 7, No. 1, p. 76-80.

Luhn, H. P. (1958). A Business Intelligence System. *IBM Journal of Research and Development* 2(4), 314–319.

Hirschman, L. (1991) Comparing MUCK-II and MUC-3. Assessing the difficulty of different tasks. *In proceedings of the Third MUC*. 25-30.

Maron, M.E. et Kuhns, J.L. (1960) On relevance, probabilistic indexing and information retrieval. *Journal of the association for comuptung Machines*. 7:216-244.

Navarro G. (2001). "A guided tour to approximate string matching". *ACM Computing Surveys* 33 (1): 31–88.

Newell, A., Shaw, J.C. et Simon, H.A. (1960) A variety of intelligent learning in a general problem solver. *In M.C. Yovits and S. Cameron (Eds.), Self-organizing systems: Proceedings of an interdisciplinary conference* (pp. 153-189). New York, NY: Pergamon Pres.

Nielsen, J. (1989). Usability engineering at a discount. In G. Salvendy & M.J. Smith (Eds.). *Designing and using human-computer interfaces and knowledge based systems* (pp 394-401). Amsterdam, The Netherlands: Elsevier Science Publishers, B.V.

Ponte, Jay M. et Croft, W. (1998). Bruce : A Language Modeling Approach to Information Retrieval. *In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. pp. 275-281.

Quinlan, J. R. (1993). *C4.5: Programs For Machine Learning. (Morgan Kaufmann Series in Machine learning)*. Morgan Kaufmann.

Ram, A. (1992). Natural Language Understanding for Information Filtering Systems. *Communications of the ACM* 35(12), 80–81.

Robertson, S.E. et Sparck Jones, K. (1976). Relevance weighting of search terms. *Journal of American for Information Science*. 27(3):129-146.

Van Meteren, R. et Van Someren, M. (2000). Using content-based filtering for recommendation. *Proceedings of MLnetECML2000 Workshop*.

Salton, G. (1969). Automatic processing of foreign language documents. *Journal of American Documentation*. 20(1):61-71.

Salton, G. (1970b). Automatic text analysis. *Science*. 168(3929):335-342.

Thomas K. Landauer, Peter W. Foltz, and Darrell Laham. (1998). Introduction to Latent Semantic Analysis Discourse Processes, 25, 259-284.

Sandel, O. Modèle d'Interface Intelligente pour Terminaux de Communication. Thèse de doctorat en informatique, UNIVERSITÉ STRASBOURG 1 - LOUIS PASTEUR. Laboratoire des Sciences de l'Image, de l'Informatique et de la Télédétection (LSIIT, UMR 7005 CNRS - ULP)Equipe de Recherche Technologique en Informatique (ERTI).

Sandvig, J. et Burke, R. (2006). AACORN: A CBR recommender for academic advising. Currently in development, School of Computer Science, Telecommunications, and Information Systems, DePaul University, Chicago,USA.

Smith et J.N. Mosier. (1986) *Guidelines for designing user interface software. MTR-10090, The MITRE corp.* Bedford, MA.

Stuart Russel et Eric Wefald. (1991). *Do the Right Thing: Studies in Limited Rationality*. MIT Press, Cambridge, Mass.

S. Muggleton et L. De Raedt. (1994). Inductive logic programming- theory and methds. *Journal of Logic Programming*. 19-20:629.

Thomas W. Malone, Kenneth R. Grant et Franklyn A. Turbak, Stephen A. Brobst, and Michael D. Cohen. (1987, May) Intelligent Information-Sharing Systems. *Communications of the ACM.* 30(5):390-402.

Wolfgang Wahlster et Alfred Kobsa. (1988). *User models in dialog systems. in ed. Wahlster and Kobsa, User models in dialog system.*, Springer Verlag, pages 4 - 34.

ANNEXE 1 – LOGICIEL DE CALCUL ET DE SIMULATION

1. Python

Python est un langage de programmation dynamique remarquablement puissant qui est utilisé dans un large éventail de domaines d'application. Nous citons certains de ses principaux traits distinctifs suivants:

- La syntaxe lisible.
- La capacité d'introspection.
- Orientation objet intuitif.
- Modularité complète, supportant les paquets hiérarchiques.
- Très haut niveau dynamique des types de données.
- Intégrable dans les applications comme une interface de script.

http://www.python.org/

Python permet d'écrire et d'exécuter du code rapidement grâce à un compilateur byte hautement optimisé et des bibliothèques de soutien.

2. Tree tagger

Le TreeTagger est un outil pour l'annotation du texte. Il a été développé par Helmut Schmid durant le projet de TC à l'Institute for Computational Linguistics de l'Université de Stuttgart. Le TreeTagger a été utilisée avec succès pour plusieurs langues telles que : allemand, anglais, français, italien, néerlandais, espagnol, bulgare, russe, grec, portugais, chinois, swahili, le latin, l'Estonie et les anciens textes français.

http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/

Mallet

Mallet est un package basé sur le langage Java. Il effectue des traitements statistiques du langage naturel, des classifications de documents, de clustering et de l'extraction d'information. Mallet comprend des outils de marquage des séquences pour des applications telles que l'extraction des entités à partir d'un texte.

http://mallet.cs.umass.edu/

3. Application R-2.13.0 pour Windows (32/64 bit)

R est un logiciel d'analyse statistique qui fournit toutes les procédures usuelles. R est un environnement pour les calculs statistiques et des graphiques. Il compile et fonctionne sur une large variété de plates-formes UNIX, Windows et MacOS. Il est similaire à l'environnement S. R peut être considéré comme une implémentation différente de S. Il possède des fonctionnalités graphiques performantes pour visualiser les données et des techniques statistiques telles que : la modélisation linéaire et non linéaire, l'analyse des séries temporelles, la classification et le clustering.

Un des avantages de R est la facilité avec laquelle la qualité de la publication des parcelles est conçue, y compris les symboles et les formules mathématiques.

http://www.r-project.org/

http://www.ceremade.dauphine.fr/~xian/Noise/R.pdf

ANNEXE 2 – DESCRIPTIONS DES COURS RECOMMANDÉS

Poly/IND6402 Interfaces humains-ordinateurs

Définition, classification et évolution des interfaces humain-ordinateur. Notions de compatibilité, accessibilité, sécurité, performance, utilisabilité, esthétique et expérience utilisateur. Méthodologie de conception centrée sur l'utilisateur. Normes, principes, critères et modèles de conception. Analyse des besoins et analyse contextuelle. Spécifications de l'utilisabilité. Modélisation, maquettage et prototypage. Styles d'interaction humain-ordinateur. Dispositifs d'entrée de données et de pointage. Présentation d'informations. Fonctionnalités de soutien à l'utilisateur. Patrons de conception. Méthodes d'inspection ergonomique. Tests d'utilisabilité. Suivi de l'utilisabilité et de l'expérience utilisateur.

UQAM/INF6104 Recherche d'informations et web

Décrire ce que sont les connaissances et l'information non structurée et le rôle qu'elles jouent dans l'organisation. Utiliser les techniques classiques de recherche d'informations et les techniques web dans Ie cadre du développement logiciel et d'activités scientifiques. Intégrer la recherche d'informations dans le développement informatique. Utiliser efficacement les informations contenues dans de grands ensembles de documents. Évaluer les différentes méthodes de recherche d'informations dans un contexte de gestion des connaissances. Les lois de Zipf et Mandelbrot. Théorie de l'information de Shannon. Les formats de métadonnées, XML. Expressions régulières: ancrages, groupement atomique, tests avant/arrière, quantificateurs avides, paresseux et progressifs. Index inversés. Arbres de suffixes. Tableaux de suffixes. Modèles booléen, vectoriels et probabilistes. Modèles de la langue. Ergonomie en recherche d'information. Hyperonymie, hyponymie, troncature, lemmatisation et thésaurus. Utilisation pratique d'un moteur de recherche dans une application avec Lucene et Snowball. Hyperliens et moteurs de recherche sur le web: PageRank et HITS. La logistique d'un moteur de recherche web. Systèmes de recommandation et filtrage collaboratif. Évaluation: précision, rappel, score F, contre-validation.

Poly/IND6409 Interfaces humains-oridnateurs

Méthodologie, normes et principes de conception et d'évaluation des interfaces humain-ordinateur. Web analytique. Web sémantique. Interfaces avec petit écran. Interfaces haptiques. Interfaces multimédia. Interfaces avec réponse vocale interactive. Interfaces des environnements virtuels. Interfaces pour des utilisateurs ayant des besoins spécifiques. Plasticité des interfaces. Interfaces de jeux vidéo. Interfaces cérébrales. Interfaces des réseaux sociaux.

Poly/IND6412 Ergonomie des sites web

Historique, développement et caractéristiques de l'Internet. Types de sites Web et caractéristiques de leurs interfaces. Langage HTML et technologies de mise en page et d'interactivité. Analyse des besoins préalable à la conception. Méthodologie de conception centrée sur l'utilisateur, normes et accessibilité. Prototypage d'interfaces. Architecture et navigation. Modes de dialogue. Conception des pages-écrans. Comportements et performance humaine liés aux sites Web. Utilisabilité des interfaces.

Poly/LOG2420 Anal. et conc. des interfaces utilisateurs

Analyse et spécification des besoins des utilisateurs. Ergonomie cognitive. Principes et règles de conception d'interface. Tests utilisateurs. Évaluation heuristique et inspection d'interface. Boîtes à outils. Système de fenêtrage. Architecture logicielle et modèle de programmation événementielle. Communication entre objets. Adaptation du processus de développement logiciel. Aide et assistance. Analyses coûts-bénéfices.

UQAM/EDM9161Interaction humain-ordinateur

Étude des modèles et des recherches sur l'interaction humain-ordinateur dans divers domaines de communication médiatisée : mondes virtuels, e-commerce, formation à distance, partage des connaissances et des ressources, technologies adaptées, intelligence et personnalisation des systèmes. Principes de conception et d'évaluation des interfaces en ergonomie cognitive.

UQAM/INF4150 Interfaces personnes-machines

Permettre à l'étudiant de concevoir des interfaces personnes-machines à l'aide de méthodes éprouvées. Matériel de support pour les interfaces. Modèles cognitifs et typologie des

utilisateurs. Classification des interfaces et paradigmes en usage. Outils d'aide à la conception des interfaces. Styles des dialogues entre les humains et la machine. Conception de l'aide contextuelle et du guide d'utilisation. Application des principes aux sites WEB. Ce cours comporte une séance hebdomadaire de deux heures de travaux en laboratoire.

UQAM/INF7510 Systèmes à base de connaissances

Introduction à l'intelligence artificielle et aux systèmes à base de connaissances. Domaines d'application. Informatique cognitive des organisations. Divers types de représentation de la connaissance. Raisonnement et moteurs d'inférences. Acquisition de connaissances. Conception de systèmes à base de connaissances: planification, méthodologie de développement, environnements de développement, langages. Systèmes à base de connaissances et systèmes d'information: interfaces intelligentes, systèmes à base de connaissances comme outil de conception, contrôle et exploitation des systèmes d'information, découverte de connaissances à partir des données. Nouvelle génération de systèmes d'information intégrant la composante cognitive. Impacts dans l'entreprise (techniques et organisationnels). Participation de l'usager et de l'expert. Rôle du cogniticien.

UQAM/INF7530 Interfaces personne-machine

Aspects cognitifs et ergonomiques. Interfaces écologiques. Modes d'interaction: dialogues, menus, icônes, graphisme, parole, multimédia. Principes directeurs de définition et de conception d'interfaces. Considérations de navigation et de gestion. Assistance aux usagers. Environnements et outils de développement. Impact organisationnel et sociologique. Applications.

UQAM/MGL830 Ergonomie des interfaces usagers

Cours relevant de l'É.T.S.

Ce cours vise à donner aux étudiants des connaissances et des méthodes pour l'évaluation expérimentale des interfaces usagers. Le cours privilégie une approche centrée sur l'agent, qui tient compte des capacités motrices, perceptives et cognitives des usagers. Durant les laboratoires, les étudiants réaliseront des tests psychophysiques sur des sujets humains, puis amélioreront des interfaces préexistantes en se basant sur les résultats de ces tests.

Concepts généraux: agent, tâche, activité, comportement, capacités, usabilité. Facteurs humains: Différents facteurs de variabilité et problèmes méthodologiques associés. Introduction aux systèmes sensoriels et moteurs. Cognition, mémoire de travail et attention. Apprentissage, actions conscientes et automatismes. Évaluation des interfaces: enquêtes, tests d'usabilité, tests psychophysiques, tests d'acceptation, vérification de cohérence, walk-through cognitifs. Conception des interfaces. Modèle de conception OAI, Keystroke Model, notation UAN. Règles d'ergonomie et guides généraux pour la conception des interfaces. Implémentations d'interfaces usagers. Interfaces graphiques (GUI) et interfaces à manipulation directe. Environnements de synthèse visuels et haptiques. Interfaces physiologiques et autres interfaces avancées.

UdM/IFT2905 Interfaces personne-machine

Concept et langages des interfaces. Programmation par événements. Modèle de l'usager. Design et programmation d'interfaces graphiques. Impact sur les multi-média, la collaboration et la communication.

UdM/CHM3404 Surfaces, interfaces et colloïdes

Chimie des surfaces, interfaces et colloïdes. Adsorption moléculaire. Couches monomoléculaires. Les propriétés physiques des systèmes colloïdaux. Polymères aux interfaces. Imagerie et profilage de surfaces. Tribologie. Nanotechnologie.

6-090-08 - Atelier de recherche en contrôle de gestion

Ce cours porte sur le processus de recherche en comptabilité ainsi que sur les principales méthodes de recherche utilisées dans ce domaine. Il cherche également à aider l'étudiant à élaborer son propre projet de recherche. Ce cours vise essentiellement à développer les habiletés de recherche des étudiants en comptabilité afin qu'ils puissent élaborer leur propre projet de recherche. Plus spécifiquement, ce cours cherche à développer des habiletés de recherche permettant :

- d'identifier et de formuler adéquatement une question de recherche,
- d'explorer la littérature pertinente à la question de recherche,
- de formuler des hypothèses ou propositions préliminaires de recherche,

- de sélectionner la méthodologie de recherche pertinente au sujet d'intérêt,
- d'identifier et de développer le ou les instruments de recherche nécessaires,
- de sélectionner l'échantillon requis,
- de recueillir et de valider les données quantitatives et/ou qualitatives recueillies,
- d'identifier des méthodes d'analyse (qualitative ou quantitative) des données à utiliser,
- d'utiliser et d'interpréter les résultats des analyses statistiques des données,
- d'utiliser le potentiel offert par Internet et
- de structurer un document de recherche tout en réfléchissant à la stratégie de publication en termes d'articles de recherche et de transfert tout en tenant compte des questions éthiques.

MAE7000 Recherche en éducation : nature et méthodologie

Ce cours vise à initier à une démarche de recherche et à lire de manière critique des rapports de recherche en termes d'élément de validité.

Diverses conceptions de la science et de la méthodologie scientifique. Les aspects épistémologiques ainsi que les aspects éthiques de la recherche et de l'intervention à des fins de recherche. Les concepts de modèles, théories, postulats, hypothèses. Types généraux de recherche; nature et application de la recherche en éducation et en formation. Recherche disciplinaire et interdisciplinaire. Survol des approches hypothético-déductives et exploratoires ou interprétatives, et mixtes. Problématique et questions de recherche ; nature et structure d'un cadre conceptuel et théorique ; formulation des objectifs ou des hypothèses. Définition et contrôle de variables. Validité interne et externe. Biais dans le processus de recherche et obstacles à la validité de la recherche.

Poly/IND3903 Projet intégrateur: systèmes d'information

L’inscription et l'abandon de ce cours-projet sont sujets à des restrictions.

Description : Projet intégrateur de conception de systèmes d'information. Modèle relationnel de données. Nomenclature des tables. Langage SQL, système de gestion de bases de données.

Méthodologie de conception de systèmes d'information pour une entreprise de production de biens et de services. Application de la réingénierie des processus. Conception et programmation des entrées et sorties de données. Logiciel de gestion de bases de données relationnelles. Technologies de l'Internet. Application de principes de gestion de projets. Intégration des habiletés personnelles et relationnelles (HPR).

Poly/INF3710 Fichiers et bases de données

Introduction aux fichiers et bases de données. Analyse de besoins : modèle entité-association. Modèle relationnel : concepts de base et algèbre relationnelle. Norme SQL (Standard Query Language) : langages de définition, de manipulation et de contrôle de données. Langage SQL enchâssé dans un langage algorithmique de programmation. Notions de contrôle d'accès concurrents et de gestion de transactions. Conception d'un schéma de base de données relationnelle : dépendances fonctionnelles et formes normales. Modèles de stockage de relations et de fichiers. Structures auxiliaires facilitant l'accès aux données : indexage et adressage dispersé.

Poly/SB100 Méthodologie de résolution de problèmes

Description des méthodologies de résolution de problèmes. Étude de la démarche systémique. Analyse des outils et méthodes de diagnostics. Analyse des outils et méthodes de prise de décisions. Analyse des méthodes d'exécution de l'action. Analyse des méthodes d'évaluation des résultats. Analyse des méthodes de rétroaction. Études de cas. Description des méthodologies de détection des pannes. Laboratoires de détection de pannes. Description de la méthodologie de recherche et développement.

UQAM/BIA1700 Organismes et environnement

Dans cette unité, les étudiants apprendront les principes de base de l'écologie des populations et de l'écologie des communautés et des écosystèmes. Ils aborderont les thèmes d'allocation des ressources, de dénombrement et de dynamique des populations, du contrôle des populations, des effets des facteurs biotiques et abiotiques, de la compétition, des intéractions prédateur-proie, de

la structure des communautés, de la succession écologique, de la productivité des écosystèmes, des chaînes trophiques, des grands biomes du monde, et des effets des perturbations humaines.

UQAM/DES1530 Projet de design I

Atelier de production ayant pour objectif d'initier les étudiants au langage de base de l'organisation spatiale. Approche des techniques de dessin et de maquette comme modes de description, de transcription et d'expression d'une idée en une forme. Exercices visant la conception et la représentation des formes dans l'espace à partir des notions de limite spatiale, de transition et de parcours.

UQAM/ECO1272 Méthodes d’analyse économique I

Ce cours a pour objectif de permettre à l'étudiant de se familiariser avec le calcul matriciel et la théorie de l'optimisation, d'acquérir une maîtrise adéquate des techniques qui leur sont associées et d'appliquer ces techniques à la résolution de problèmes économiques. Méthodes de résolution des modèles économiques linéaires. Application de ces méthodes aux problèmes de l'allocation sectorielle de la production et de l'obtention des niveaux d'équilibre des variables dans les modèles économiques simples. Introduction des concepts de l'analyse marginaliste en économie et leur application aux fonctions d'utilité, de production et de coût. Analyse des fonctions d'offre et de demande et calculs d'élasticités. Techniques d'optimisation pour économistes. Recherche des valeurs extrêmes des fonctions d'une ou de plusieurs variables, en l'absence de contraintes. Application de ces techniques à la résolution des problèmes de la maximisation du profit dans les entreprises concurrentielles et en situation de monopole. Recherche des valeurs extrêmes en présence de contraintes; application à la minimisation des coûts des entreprises et à la maximisation de l'utilité des consommateurs. Caractéristiques des fonctions de production: homogénéité et rendements d'échelle. Cours avec séances de travaux pratiques.

UQAM/ENV7140 Principes de gestion intégrée des ressources

Ce cours vise à procurer aux étudiants des outils conceptuels et pratiques permettant de gérer les ressources naturelles dans une optique de développement durable, par la prise en compte du caractère multifonctionnel des ressources, de la diversité d'activités soutenues par l'environnement, de l'intérêt collectif actuel et futur, de l'intégration des préoccupations

environnementales à toutes les étapes de décision et d'une démocratisation de la prise de décision. Historique, fondements et évolution de la gestion des ressources et de l'environnement. Diversité des intervenants et des modes de participation à la prise de décision. Utilisation des données biophysiques. Inventaire des options et élaboration des scénarios. Méthodes d'aide à la décision multicritères. Application à des domaines et des problématiques telles que la gestion des ressources forestières, le développement rural, la gestion de l'eau, etc. Ce cours comporte des sorties sur le terrain et des séances de laboratoire.

UQAM/GHR6700 Gestion de l'hébergement

Ce cours vise à rendre l'étudiant capable de: analyser et caractériser un système donné, qu'il s'agisse du hall et de la réception d'un hôtel, de la conciergerie ou des étages (chambres et suites); évaluer l'efficacité des processus de travail, la qualité des services offerts à la clientèle et la rentabilité des opérations d'un département donné; planifier, organiser et diriger un département de l'hébergement en tenant compte des problématiques identifiées et de la mission déjà définie. Études de cas et mises en situation favorisant le réinvestissement des compétences managériales pertinentes à la gestion de l'hébergement. Analyse de la cohérence des décisions des gestionnaires quant à la définition des produits et des services, de l'interface avec la clientèle, des processus de travail et de contrôle, de l'utilisation des ressources et de l'espace, des interrelations entre les départements et de la mission d'une entreprise donnée.

UQAM/JUR2508 Système et documentation juridiques canadiens

Système et processus législatifs, organisation juridictionnelle et corpus de doctrine au Canada et au Québec. Méthode de suivi de l'évolution des textes législatifs et réglementaires (publication, entrée en vigueur, modification, abrogation) et du cheminement des appels.

Étude des usages des principaux instruments et ouvrages de référence, de repérage et de suivi bibliographiques. (Statuts du Canada, Statuts révisés du Canada, Lois du Québec, Lois refondues du Québec, Tableau des modifications des lois, Gazettes, Canadian Abridgment, Annuaire de jurisprudence du Québec.). Examen de la structure et des contenus des banques de données des fournisseurs canadiens et québécois. (Justice Canada, Lexis Nexis, ECarswell, Société québécoise d'information juridique, Répertoire électronique de jurisprudence du Québec).

Typologie des tâches de recherche documentaire. Stratégies de recherche documentaire : détermination des objets de recherche, de l'horizon temporel et des aires de recherche, choix des outils appropriés à un objet de recherche donné, techniques d'interrogation des bases de données. Processus et procédés permettant d'établir l'état du droit relativement à une question donnée.
Règles formelles de rédaction des travaux juridiques, de formulation des renvois et de présentation de bibliographies Ateliers en bibliothèque et en ligne. Production d'une recherche documentaire assortie et intégrée à un texte. Cours assorti d'un atelier d'une heure en bibliothèque ou en laboratoire.

UQAM/MSL9001 Méthodologie de la recherche en muséologie (3 cr.)

Ce séminaire de méthodologie vise à favoriser une démarche rigoureuse de recherche en muséologie en privilégiant la compréhension des étapes cognitives de conception, d'élaboration et de validation d'une recherche, ainsi que l'élaboration de problématiques critiques intégrant au sein de celles-ci des catégories d'analyse provenant de différents paradigmes.

Ce séminaire théorique enclenche un apprentissage méthodologique permettant la réalisation de projets de recherche et la définition du cadre général des méthodologies de la recherche liées aux trois axes du programme. Cet apprentissage est effectué par le biais d'une étude du processus de recherche qui comprend: énoncé de problèmes, formulation des hypothèses ou des questions de recherche, maîtrise des instruments de travail, présentation détaillée des différentes étapes d'un projet de thèse de doctorat. Le contenu du séminaire pourra varier en fonction des caractéristiques des sujets des candidats inscrits.

UQAM/INF7870 Fondements logiques de l'informatique

Revue de la logique propositionnelle et du premier ordre. Logique temporelle et logique modale. Logique floue. Déduction naturelle. Tableaux sémantiques. Heuristiques et tactiques de preuves. Applications.

UQAM/LIN1400 À la découverte du langage

Ce cours vise à sensibiliser les étudiants aux concepts fondamentaux relatifs au langage, particulièrement dans un contexte scolaire. Les thèmes abordés incluent: la nature du langage

humain; le langage comme outil de communication; le langage comme phénomène social; les sons et les systèmes sonores; l'écriture et les systèmes orthographiques; la structure des mots, des phrases et des textes; aspects sémantiques et pragmatiques du langage; langage et paralangage. Examen de la diversité des langues afin de permettre une meilleure compréhension de leurs ressemblances et de leurs différences par rapport au français.

UQAM/LIN3557 Sémantique II

Ce cours a comme objectif de préciser les propriétés des analyses sémantiques et de présenter différentes théories proposées dans le cadre de la grammaire générative: sémantique générative, sémantique interprétative, sémantique configurationnelle. La capacité descriptive et explicative de ces théories sera évaluée à la lumière d'un modèle grammatical restrictif. Enfin, le rôle de l'analyse sémantique à l'intérieur de la description grammaticale sera abordé.

UQAM/LIN8203 Sémantique 1

Ce cours constitue une introduction aux approches formelles en sémantique. On présentera un modèle visant à construire l'interprétation des énoncés à partir de la contribution sémantique des éléments qui les composent. Les thèmes fondamentaux de la sémantique seront étudiés: la distinction entre le sens et la référence, les relations prédicat-argument, la sémantique lexicale, la quantification, le temps, l'aspect et la modalité. Divers outils logiques et formels servant à l'étude empirique des phénomènes sémantiques seront introduits. On verra également comment divers travaux de sémantique formelle ont contribué à certains développements théoriques en syntaxe.

UQAM/PHI3512 Sémantique et pragmatique

Étude des développements majeurs que la sémantique a connus au cours du XXe siècle, présentation des principaux concepts et problématiques de la théorie du sens, de la référence et de la vérité. Étude des principales problématiques pragmatiques (actes de langage, indexicalité, sens non littéral) et de leurs effets sur les théories sémantiques classiques. On s'attachera en particulier aux relations entre les approches sémantique et pragmatique du langage, par l'examen de concepts comme ceux de croyance, d'intention ou de présupposition.

UQAM/PHI4352 Logique informelle

Les objectifs généraux du cours sont d'initier à la logique comme moyen pratique d'agir dans la décision et dans l'argumentation. Distinguer entre la dimension logique du langage relativement à ses dimensions non logiques (sémantique, pragmatique, etc.). Les objectifs spécifiques sont d'initier à la logique classique interpropositionnelle, à la logique classique intrapropositionnelle et à la logique de la décision. Opposer les arguments logiques aux erreurs logiques (syntaxiques, sémantiques, pragmatiques et rhétoriques) les plus répandues.

Sensibilisation à l'importance du langage dans le comportement humain et à la place de la logique dans le langage. Introduction à la logique interpropositionnelle. Introduction à la logique intrapropositionnelle. Introduction aux sophismes extra-logiques. Introduction à la logique de la décision. Récapitulation générale et retour sur l'importance du langage dans le comportement humain et sur la place de la logique dans le langage.

UdM/LNG1000 Introduction aux langages formels

Étude des langages formels en tant qu'outils de représentation des structures sémantiques, syntaxiques et morphologiques de la langue

UdM/LNG2080 Sémantique du français

Représentations sémantiques (réseaux sémantiques), représentations syntaxiques (arbres de dépendance) et leur correspondance (règles sémantiques) dans la description du français.

UdM/LNG3050 Sémantique : théorie et description

Sémantique lexicale : décomposition et primitifs sémantiques. Étude des collocations. Structure communicative des énoncés.

UdM/ORA1534 Langage 1

Acquérir une compréhension de la morphologie, du lexique et de la sémantique.

SYS841 Systèmes experts

Faire comprendre à l'étudiant l'état actuel de la technologie des systèmes experts (SE) et pressentir les développements à venir. Lui permettre de choisir une application viable et de comparer sur une base rationnelle les outils logiciels nécessaires à son développement.

Principes de base et définitions. Architecture des SE. Représentation des connaissances: logique des prédicats, règles de production, réseaux sémantiques et objets structurés. Mécanismes d'inférence. Stratégies de contrôle dans les SE basées sur des règles de production: chaîne avant et arrière, etc. Typologie des SE avec exemples d'application. Méthodologie de construction des SE: choix de l'application et de l'outil appropriés, transfert de l'expertise et étapes de réalisation. Outils logiciels: présentation succincte de LISP, de Prolog et des outils de développement avec une description plus approfondie de l'un d'eux. Avantages et limites des SE. Réalisation d'un prototype de SE de diagnostic avec l'outil décrit.

PHI3508 Logique intermédiaire

Les méthodes formelles les plus couramment utilisées par la philosophie contemporaine et des notions de métalogique. En particulier, les notions de modèle et de système axiomatique, en calcul des énoncés puis en calcul des prédicats. Les liens entre syntaxe et sémantique à travers les résultats classiques: théorème de déduction, théorème de complétude, décidabilité et indécidabilité. Aperçu de logiques non classiques: la logique modale et son interprétation sémantique, la logique intuitionniste, etc. Enfin, sur le plan pédagogique, on prévoira en classe un certain nombre d'exercices d'application.

Poly/LOG4420 Conception de sites web dynam. et transact.

Conception de sites web complexes pour la génération dynamique de contenu et la gestion d'interactions avec les utilisateurs. Présentation générale de l'architecture du web et du protocole HTTP (HyperText Transfer Protocol). Structure d'un document HTML (HyperText Markup Language). Mise en forme d'un document HTML par l'utilisation de CSS (Cascading Style Sheet). Paradigmes de conception propres aux systèmes web. Programmation du côté serveur. Gestion d'une session sur un site web. Éléments de sécurité pour les sites web. Présentation du

format XML (Extended Markup Language) et du langage de transformation de documents XSL (Extended Stylesheet Language). Programmation du côté client par le biais de scripts exécutés par le navigateur web. Interface avec une base de données relationnelle. Notions de performance et de sécurité. Notions de validation et de test de sites web dynamiques et transactionnels.

ORH3000 Méthodes de recherche appliquées à la gestion des ressources humaines (3 cr.)

L'objectif du cours est de permettre à l'étudiant d'acquérir des connaissances de base sur les méthodes de recherche pertinentes au domaine de la gestion des ressources humaines et de les mettre en application dans le cadre d'exercices pratiques.

Le cours présente une définition des concepts de recherche scientifique, les principales étapes d'une recherche, les objectifs et les types de recherche pertinents au domaine de la gestion des ressources humaines. On y traite également de l'élaboration d'une problématique de recherche, du développement des théories et des hypothèses de recherche, des types de stratégies et de devis de recherche, de l'échantillonnage, des méthodes de collecte de données, de la codification des données, des méthodes d'analyse et de la présentation d'un protocole de recherche.

Interfaces et scénarisation

Apprentissage des principes de scénarisation interactive et du processus de conception et d'évaluation des applications de communication informatisée, en utilisant diverses théories ergonomiques.

PSY5890 Utilisations de l'ordinateur en psychologie (

Les objectifs de ce cours sont les suivants: Réaliser l'importance et l'utilité de l'ordinateur dans son champ de formation. Comprendre l'essentiel de la nature et du fonctionnement d'un ordinateur. Se familiariser aux utilisations multiples de l'ordinateur dans le contexte de sa formation à la recherche et à l'intervention. Acquérir une compétence dans une application importante de l'utilisation de l'ordinateur.

Nature et utilités de l'ordinateur et du microordinateur: structure et types d'ordinateurs, langage de programmation et niveaux de langages. Usages de l'ordinateur a) en recherche: étude et simulation de processus psychologiques, intelligence artificielle, contrôle d'expérience, collecte et analyse de données, b) en psychologie de l'éducation: ordinateur et développement des habiletés à

penser; résolution de problèmes, visualisation, organisation de l'information et des opérations... Travail pratique selon les intérêts et les connaissances antérieures de l'étudiant, installation de périphériques ou de cartes sur un ordinateur, construction d'un site Web ou d'un programme simple, apprentissage d'un progiciel chiffrier (par exemple: Excel) ou base de données bibliographiques. Présentation en classe du travail avec application multimédia.

Administration des informations et des données

Les différentes fonctions de gestion des données: modélisation des informations d'entreprise; définition des données utiles à l'entreprise; normalisation de la définition et de l'utilisation des données; structuration, création, contrôle et documentation des bases de données; planification de l'extension des données et de l'évolution du système de gestion des données; contrôle du dictionnaire des données; formation en informatique des utilisateurs. Les outils de conception et de contrôle disponibles. Le rôle de l'administrateur dans l'entreprise. Conflits d'intérêt entre les utilisateurs. Relations entre l'administration et les utilisateurs.

Internet et création de pages Web

Introduction à la création de pages Web interactives. Mise en page, ergonomie et intégration de contenus multimédia.

Information et sites Web

Historique du Web. Normes du W3C. Méthodologie de développement. Langage XHTML. Intégration de contenu multimédia. Feuilles de style CSS. Ergonomie et accessibilité. Design et graphisme. Intégration de JavaScript. Référencement. Sécurité.

Sites Web avancés avec Frontpage

Création de sites web avancés avec Frontpage. Mise en place de contenus dynamiques dans les sites Web. Connexion à une base de données. Gestion de formulaires.

LNG2240 Pragmatique

Étude des conditions d'utilisation du langage. Description des actes de parole, des motivations et des réactions des interlocuteurs, des types de discours. Implications en traduction.

DES7520 Composants humains, anthropométrie et activités des utilisateurs

Vu l'importance des enjeux liés à la santé et à la sécurité des personnes, le cours aborde l'acte de conception en mettant l'accent sur l'utilisateur. Description des caractéristiques des utilisateurs dans l'industrie du transport. Que ce soit à partir de données issues de laboratoires ou encore issues d'analyses de terrain, le cours présente l'ensemble des techniques scientifiques utiles afin de connaître avec précision les caractéristiques physiologiques et psychologiques des utilisateurs. Le cours présente des méthodes d'enquête afin de documenter des problématiques complexes de confort à court et à long terme. Il présente les méthodes optimales d'utilisation de mannequins anthropométriques virtuels lors du développement de formes ainsi que les autres techniques de recueil de données anthropométriques.

UQAM/DES7530 Normes et sécurité dans les transports

Présenter les principes guidant la sécurité dans les transports. Ce cours apporte des connaissances sur les différentes sources de réglementation en matière de transport et de gestion des flux (critères en matière de signalisation, gestion de la circulation - individus, foules et véhicules, manipulation de matières dangereuses, ...). Mise en perspective des fondements et approches d'analyse de l'erreur humaine et de la fiabilité opérationnelle. Une emphase est portée sur le modèle de résolution de problème de Rasmussen. Notions de sécurité et variabilité circonstancielle et de marge de manoeuvre. La prévention des incidents et des accidents dans des situations dynamiques. Présentation des modèles de l'arbre des causes comme point de départ à la conception. Utilité et limite des tests d'impact. Catastrophes naturelles et industrielles: les enseignements qu'on peut en tirer. Vandalisme et résistance aux changements.

PSY9433 Herméneutique et psychothérapie

Cours sous forme de séminaire. Présenter et élaborer, en référence à la philosophie de Hans Georg Gadamer, les fondements d'une conception herméneutique de la pratique de la psychothérapie. Interroger le mode de mise en culture de celle-ci dans le contexte de l'institution moderne et actuelle de l'intention thérapeutique, en psychologie clinique. Étude et

approfondissement, du point de vue de l'herméneutique (v.g. Hans Georg Gadamer) des phénomènes comme ceux de la psychopathologie et de la souffrance. Interrogation de l'horizon, de la portée, et des limites du discours normatif en contexte d'intervention psychologique. Élaboration des principes et des réflexions qui conditionnent l'articulation existentielle de la méthode, de la pratique et de la théorie de la psychothérapie. Rapprochement et mise en contraste des conceptions existentielles et médicales (par exemple) de l'affect et de l'angoisse, de l'autorité et de la liberté critique, de la technique, de sens de l'interprétation. Étude du caractère dialogique et narratif de l'espace psychothérapeutique. Examen de concepts tels ceux de corporéité et vie, de symptôme, d'identité et de subjectivité, d'événement et de compréhension, d'influence et d'autonomie.

Ergonomie avancée

Et rôle de l'ergonomie dans une approche multidisciplinaire de conception de produits et de systèmes de production de biens et services. Méthodologies de conception ergonomique. Outils informatiques pour intégrer l'ergonomie lors de la conception. Pratiques innovantes en gestion de la production et problématiques de santé et de sécurité du travail. Chronobiologie, rythmes biologiques et critères de conception des horaires de travail. Conception de l'environnement physique : confort et contrainte thermique; environnement visuel et éclairage; dimensionnement des espaces de travail, des équipements et des machines. Vigilance, attention, détection du signal; conscience de la situation et prise de décision; charge mentale de travail; fiabilité et erreurs humaines. Formation des travailleurs.