

## Chapitre 4

# Connexionisme et génétique pour la recherche d'information

### 4.1. Introduction

Il est maintenant avéré qu'au centre de la problématique de la recherche d'information s'impose un investissement considérable et porteur pour l'intégration de fonctionnalités d'adaptation des modèles et stratégies de recherche aux profils des utilisateurs.

L'extraction instructive d'informations à partir des interactions entre système de recherche d'information et utilisateurs a pour objectif d'améliorer les performances du processus de recherche en termes de qualité et coût des réponses fournies. Sous l'angle de la mise en œuvre, l'apprentissage émerge comme une dimension incontournable dans le contexte actuel de développement des systèmes de recherche d'information.

Théories déjà anciennes, le connexionisme et plus récemment les algorithmes génétiques constituent des supports formels offrant un cadre opportun pour la modélisation de l'apprentissage dans un système de recherche d'information. De nombreuses approches ont exploré l'utilisation des réseaux de neurones et des algorithmes génétiques pour appréhender le problème de l'apprentissage. Ces approches présentent des caractéristiques communes dans le principe mais beaucoup en différent quant aux objectifs et moyens mis en œuvre.

Fondamentalement, les réseaux de neurones regroupent une classe de modèles dont l'objectif est d'imiter les fonctions humaines de mémorisation associative de l'information et de l'apprentissage. Sous l'angle de la recherche d'information, ceci ouvre des voies vers la représentation associative d'unités documentaires d'une part, et d'adaptation de la structure et fonctionnement du réseau d'informations en rapport avec les jugements de pertinence des utilisateurs d'autre part.

Par ailleurs les algorithmes génétiques font partie intégrante de la classe des algorithmes évolutifs fondés sur la simulation du processus d'évolution et d'adaptation humain dans les milieux naturels. Ces algorithmes sont adaptés à la résolution de problèmes dont l'espace de recherche est caractérisé par un grand nombre de dimensions. C'est justement un des aspects de la problématique en recherche d'information.

Au-delà des spécificités des réseaux de neurones et des algorithmes génétiques, les travaux d'application en recherche d'information se heurtent à une difficulté commune liée à la modélisation et qui se traduit par la question : comment transposer et exploiter les concepts et méthodes issus de ces théories au cadre de la modélisation de l'information de manière générale et de l'apprentissage plus particulièrement ?

L'objectif de ce chapitre est d'isoler au sein du processus de recherche d'information, les problèmes susceptibles d'être résolus par une approche fondée sur les réseaux de neurones ou sur les algorithmes génétiques puis de les illustrer à travers des modèles proposés par des auteurs du domaine. Ce chapitre est organisé comme suit :

- la section 4.2 introduit une présentation globale du processus de recherche d'information au travers la dimension utilisateur ;
- la section 4.3 présente les concepts du modèle connexioniste puis en introduit les modèles de recherche d'information les plus représentatifs de la littérature ;
- la section 4.4 fait le point sur les algorithmes génétiques en introduisant leur cadre d'utilisation en recherche d'information ;
- en conclusion (section 4.5), nous dressons un bref bilan des acquis du domaine.

#### **4.2. Evaluation de la pertinence en recherche d'information**

Indépendamment des modèles théoriques sous-jacents et des modes d'expressivité du besoin en information, la finalité évidente d'un processus de recherche d'information est de sélectionner l'information pertinente pour un utilisateur.

Ceci rejoint la problématique pendante de l'évaluation de la pertinence. Notion ambiguë, peu formelle et mal identifiée, elle est pourtant au centre de l'évaluation des performances d'un système de recherche d'information. L'utilisateur étant seul juge de la qualité des réponses du système, il devient de fait, une composante privilégiée à intégrer dans la conception des systèmes de recherche d'information.

A ce propos, Green [GRE 95] considère que l'utilisateur est le seul juge de ce qui est pertinent mais que d'un autre côté, il émet son jugement en l'absence de connaissances lui permettant d'évaluer la pertinence réelle des documents sélectionnés par le système. Ceci introduit alors la dichotomie récurrente entre pertinence utilisateur et pertinence système. Cette dichotomie étant à l'encontre des performances, il s'impose alors une recherche collaborative utilisateur – système à même de réduire les décalages de mesures de la pertinence.

A cet effet, nous introduisons l'impact et domaine d'intégration de l'utilisateur dans le processus de recherche puis développons l'idée de l'adaptation des systèmes comme réponse à la problématique de l'évaluation de la pertinence.

#### **4.2.1. *L'utilisateur au centre du processus***

En recherche d'information traditionnelle, le processus classique consiste à formuler en un épisode de recherche, l'expression d'un besoin en information au travers le langage de requêtes du système puis de l'apparier aux représentants des documents.

L'utilisateur intervient au terme du processus pour accepter ou rejeter les documents sélectionnés. Cette vision passive de l'utilisateur a montré de nombreuses limites [CRO 79, SAL 89].

D'autres approches ont investi l'idée d'intégrer l'utilisateur dans le processus de recherche. Les voies sont diverses :

- identification et évaluation de critères jouant un rôle dans l'évaluation de la pertinence utilisateur [BAR 94, HOW 94] ;
- étude et recherche de modes d'interaction efficaces utilisateur-système [BRU 94, DEN 97] ;
- techniques de modélisation de l'utilisateur [BRA 94, RIC 83].

Une autre voie consiste à intégrer l'utilisateur par reproduction du système cognitif humain dans le fonctionnement interne du système de recherche d'information. L'idée dominante est ainsi de reproduire les modèles humains de représentation, sélection et apprentissage de l'information. C'est précisément dans

ce cadre que s'inscrivent la modélisation connexioniste et génétique pour la recherche d'information.

#### **4.2.2. Apprentissage : émergence du besoin**

L'étude de la notion de pertinence renvoie sans doute à l'analyse des interactions système utilisateur au travers différents angles :

- l'expression du besoin en information : l'objectif étant d'arriver à une description unique de l'objet de la recherche ;
- la sélection de documents : l'objectif étant d'unifier les critères de pertinence des documents ;
- la représentation de l'information : l'objectif étant de projeter le contenu documentaire à la vision ponctuelle d'un unique utilisateur.

De cette perception, émerge le besoin d'apprentissage bi-univoque utilisateur système. C'est dans ce contexte que sont apparues les méthodes de réinjection de la pertinence (réinjection de nouveaux critères de pertinence découverts lors d'une étape de recherche antérieure) [BUC 94, ROB 95, ROC 71] et de propagation de pertinence dans le cas de documents web [DEA 99, KLE 98].

La présentation qui suit se focalise sur la combinaison de nombreuses sources d'apprentissage à partir de l'utilisateur :

- simulation du mode humain d'accès,
- sélection et apprentissage de l'information au travers l'utilisation de réseaux de neurones,
- optimisation des éléments d'interaction utilisateur – système selon un processus génétique.

#### **4.3. Le modèle connexioniste**

Les réseaux de neurones supportent de nombreux modèles dont l'objectif est d'imiter les fonctions de représentation et traitement de l'information du système nerveux humain. Un réseau de neurones est composé de nœuds et de liens. A chaque nœud sont associées des entrées et sorties valuées. A chaque lien est associé un poids traduisant le degré d'interconnexion des nœuds qu'il relie. Le fonctionnement du réseau est basé sur la propagation des signaux d'activation depuis les entrées jusqu'aux sorties.

L'une des propriétés fondamentales d'un réseau de neurones est la dynamique de ses états. Celle-ci traduit l'apprentissage du réseau par changement de son

comportement grâce à l'évolution des poids de ses connexions en cours du temps. De nombreuses approches de l'apprentissage ont été proposées. On y présente généralement des règles de modification de poids telles que les règles de Hebb [HEB 49] et rétropropagation du gradient [LEC 86, MCL 86].

Les systèmes de recherche d'information basés sur l'approche connexioniste utilisent les fondements des réseaux de neurones tant pour la modélisation des unités textuelles que pour la mise en œuvre du processus de recherche d'information. Le modèle offre en effet des atouts intéressants pour la représentation des relations entre termes (synonymie, voisinage, etc.), entre documents (similitude, référence, etc.) et entre termes et documents (fréquence, poids, etc.). En outre, sa propriété intrinsèque d'apprentissage permet de supporter de manière inhérente à son fonctionnement, le processus de reformulation de requête et/ou réinjection de pertinence utilisateur.

Il n'existe pas une représentation unique d'un réseau de neurones pour la recherche d'information. Cependant, l'architecture la plus répandue est celle basée sur l'interconnexion de couches représentant les éléments d'un système de recherche d'information. Le constructeur identifie précisément :

- les différentes couches du réseau :  $C_i$  (couche  $i$ ),
- les neurones de chaque couche :  $n_{ij}$  (neurone  $i$  de la couche  $C_j$ ),
- la fonction de sortie de chaque neurone :  $f_{ij}$  (fonction de sortie du neurone  $n_{ij}$ ),
- les liens et leurs poids :  $l(n_{ij}, n_{kl})$  (poids du lien entre les neurones  $n_{ij}$  et  $n_{kl}$ , vaut 0 si les deux neurones ne sont pas connectés),
- la couche d'entrée, celle qui reçoit l'entrée du réseau,
- la couche de sortie : celle qui indique le résultat du réseau.

Le premier modèle connexioniste pour la recherche d'information a été présenté par Belew [BEL 89]. D'autres modèles basés sur une architecture à couches [CRO 94] ou implémentant l'approche probabiliste [CRE 94] ont été développés et expérimentés sur des collections de test moyennes. Dans le but d'illustrer l'application des concepts du modèle connexioniste à la recherche d'information, nous détaillons dans les paragraphes suivants les modèles PIRCS [KWO 89, KWO 99] et MERCURE [BOU 92, BOU 97].

#### 4.3.1. *Le modèle PIRCS*

Le modèle construit dans Kwok [KWO 99] utilise trois couches interconnectées : couche requête, couche termes et couche documents (voir figure 4.1).

Les connexions sont bidirectionnelles et de poids assymétriques. L'approche de Kwok est fondée sur l'idée que les requêtes et documents sont des composants conceptuels rapprochés, en ce sens qu'ils sont tous deux représentants de concepts. Sur cette base, il reprend des éléments du modèle probabiliste et calcule la pertinence d'un document  $RSV(q,d)$  (*Relevance Status Value*, expression communément utilisée dans la littérature anglaise) comme une combinaison de deux valeurs de pertinence : la première est produite par le document, soit d, et la deuxième produite par la requête, soit q :

$$RSV(q, d) = \alpha RSV_d + (1 - \alpha) RSV_q \quad [4.1]$$

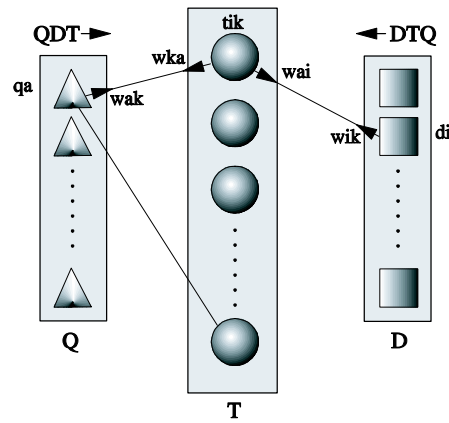


Figure 4.1. Le modèle de réseau Kwok [KWO 99]

$RSV_d$  représente la pertinence focalisée sur le document, obtenue par propagation des signaux d'activation depuis les neurones de la couche requête jusqu'aux neurones de la couche document :

$$RSV_d = \sum_k S\left(\frac{q t f_k}{L_q}\right) w_{dk} \quad [4.2]$$

$RSV_q$  représente la pertinence focalisée sur la requête, obtenue par simulation de la pertinence du document d puis propagation des signaux d'activation depuis les neurones de la couche document jusqu'aux neurones de la couche requête :

$$w_{dk} = \log \left( \frac{tf_k}{L_d - tf_k} \frac{Nw - L_d - F_k + tf_k}{F_k - tf_k} \right) \quad [4.3]$$

$$RSV_q = \sum_k S \left( \frac{q tf_k}{L_q} \right) w_{qk} \quad [4.4]$$

$$w_{qk} = \log \left( \frac{q tf_k}{L_q - q tf_k} \frac{Nw - F_k}{F_k} \right) \quad [4.5]$$

où :

$tf_k$  : fréquence du terme  $k$  dans le document  $d$

$qtf_k$  : fréquence du terme  $k$  dans la requête  $q$

$L_d = \sum_k tf_k$  : longueur du document  $d$

$S$  : fonction sigmoïde

$L_q = \sum_k q tf_k$  : longueur de la requête  $q$

$Nw = \sum_k F_k$  : nombre de termes dans la collection

Ce processus est itéré pour chaque neurone document  $d$ .

En outre, un processus de reformulation de requête est implémenté sur la base d'un algorithme d'apprentissage du réseau. A cet effet, les signaux d'activation sont propagés depuis les nœuds documents vers les nœuds termes permettant ainsi de réviser les poids des termes et/ou ajouter de nouveaux termes à la requête.

#### 4.3.2. Le modèle MERCURE

MERCURE [BOU 92, BOU 97] est un système de recherche d'information modélisé par un réseau connexioniste à trois couches interconnectées. Les requêtes, documents et termes sont représentés par des nœuds reliés entre eux. On distingue trois types de liens :

– lien *terme-document* représentant un lien descriptif dont le poids est calculé selon la formule inspirée d'Okapi [ROB 94] :

$$d_{ij} = \frac{tf_{ij} * (h_1 + h_2 * \log(\frac{N}{n_i}))}{h_3 + h_4 * \frac{dl_j}{\Delta d} + tf_{ij}} \quad [4.6]$$

où :

$h_i$  : paramètre dépendant de la collection de documents

$tf_{ij}$  : fréquence du terme  $t_i$  dans le document  $d_j$

$N$  : nombre de documents dans la collection

$dl_j$  : taille du document  $d_j$  en nombre de termes, sans les mots vides

$\Delta d$  : taille moyenne des documents de la collection

– lien *terme-requête* représentant un lien descriptif dont le poids est calculé selon la formule :

$$q_{ui}^{(s)} = \begin{cases} \frac{nq * qtf}{nq - qtf} & \text{si } (nq > qtf) \\ qtf & \text{sin on} \end{cases} \quad [4.7]$$

où :

$qtf$  : fréquence d'un terme dans une requête

$nq$  : nombre de termes dans la requête

$q_{ui}^{(s)}$  : poids du terme  $i$  dans la requête  $u$  à l'étape 0 de la recherche,  $s = 0$  car cette formule intervient à l'initialisation du processus d'évaluation de requête

– lien *terme-terme* représentant un lien de cooccurrence dont le poids est calculé selon la formule :

$$C_{ij} = \alpha * Co(t_i, t_j) \quad [4.8]$$

où :

$Co(t_i, t_j)$  : mesure de cooccurrence entre les termes  $t_i$  et  $t_j$

La fonction d'évaluation mesure la similitude entre la requête à l'entrée du réseau et documents. Elle est mise en œuvre grâce à un processus de transfert des activations de la couche requête vers la couche documents. A la fin du processus, les nœuds sont classés par ordre décroissant de leurs valeurs d'activation et les documents associés sont présentés à l'utilisateur. Le processus est réalisé selon les étapes suivantes.

1. Indexation de la requête et représentation sous la forme :



$$Q_u^{(s)} = (q_{u1}^{(s)}, q_{u2}^{(s)}, \dots, q_{uT}^{(s)}) \quad [4.9]$$

Les poids des termes dans la requête sont affectés aux liens requête-termes.

2. Déclenchement de l'évaluation à partir du nœud requête, en envoyant un signal de valeur 1 à travers les liens requête-termes.

3. Calcul d'une valeur d'entrée et valeur de sortie à chaque nœud terme :

$$In(t_i) = q_{ui}^{(s)} \quad [4.10]$$

$$Out(t_i) = g(In(t_i)) \quad [4.11]$$

4. Transmission des signaux vers la couche documents. Chaque nœud document calcule une entrée selon la formule :

$$Ind(d_i) = \sum_{i=1}^T Out(t_i) * d_{ij} \quad [4.12]$$

puis une valeur d'activation selon la formule :

$$Out(d_i) = g(Ind(d_j)) \quad [4.13]$$

$T$  : nombre total de termes d'indexation

5. Tri des documents répondant à la requête selon l'ordre décroissant de leur valeur d'activation.

Deux modes de reformulation automatique de requête sont proposés dans le système :

– *reformulation directe* : consiste à ajouter à la requête initiale les termes les plus actifs, atteints par transfert d'activation à partir des termes de la requête, et ce à la première itération ;

– *reformulation indirecte* : reformulation basée sur les résultats de recherche. A cet effet, deux stratégies sont mises en œuvre. La première est la rétropropagation de la pertinence ; elle consiste essentiellement à rétropropager les activations à partir d'une sortie désirée déterminée à partir du jugement de pertinence de l'utilisateur. Le processus de propagation inverse, de la couche documents vers la couche d'entrée, permet de corriger les poids des liens requêtes-termes. La seconde consiste en

l'adaptation de l'algorithme de rétro-propagation du gradient. Le modèle a été validé sur la collection TREC [BOU 99a].

#### 4.3.3. Synthèse

Le modèle connexionniste constitue un support formel permettant de couvrir de nombreux aspects de la recherche d'information. Plus précisément, ce modèle est caractérisé par :

- la capacité de modélisation des concepts par association d'éléments (termes, documents, liens termes-documents, liens termes-termes...);
- la pertinence du principe d'appariement requête-document puisqu'il est analogue au processus humain de sélection et recherche d'information ;
- la capacité d'apprendre sous l'effet des signaux de renforcement provenant de l'utilisateur.

Les modèles développés dans la littérature ont apporté leur preuve de performances sur divers volumes de collections. Cependant, la mise en œuvre du modèle pose des problèmes non négligeables liés à la définition de la structure du réseau et coût de convergence de l'algorithme d'apprentissage.

Notons à juste titre que ce coût est un facteur à considérer avec d'avantage d'intérêt dans le contexte d'un apprentissage basé sur les interactions avec l'utilisateur.

#### 4.4. Génétique et recherche d'information

Les algorithmes génétiques font partie de la classe des algorithmes évolutifs [BAE 91, GOL 89, HOL 75, KOZ 92]. Les techniques d'algorithmique évolutive ont attiré une attention considérable en raison des potentialités qu'elles offrent pour la résolution de problèmes complexes. Ces techniques basées sur le principe puissant de *survie du meilleur*, modélisent les phénomènes naturels liés à la génétique darwinienne ; elles constituent une catégorie intéressante d'heuristiques de recherche et d'optimisation modernes.

Les algorithmes évolutifs sont caractérisés par :

- la manipulation d'une population d'individus représentant les solutions candidates au problème posé ;
- l'évaluation de la qualité des individus grâce à une fonction d'adaptation ;
- la détermination d'une stratégie de sélection des individus d'une génération à une autre ;

- l'application d'opérateurs de transformation d'individus entre générations.

Dans la suite, nous présentons brièvement les principes de l'algorithmique génétique puis nous présentons une synthèse des travaux d'application à la recherche d'information.

#### 4.4.1. *Qu'est-ce qu'un algorithme génétique ?*

Un algorithme génétique [GOL 89, HOL 75] a pour but de faire évoluer un ensemble de solutions candidates à un problème posé vers la solution optimale. Cette évolution s'effectue sur la base de transformations inspirées de la génétique, assurant de génération en génération, l'exploration de l'espace des solutions en direction des plus adaptées.

Considérons un problème d'optimisation donné ; sa résolution sous l'angle de la génétique se résume par la succession des étapes suivantes :

- *modélisation* : consiste à identifier la description d'une solution candidate à travers un ensemble de caractéristiques, puis d'associer une fonction analytique permettant de mesurer sa capacité à résoudre le problème posé ;
- *génération de la population initiale* : consiste à créer de manière aveugle ou guidée, par application d'heuristiques, la population initiale d'individus ;

```

Début
t := 0
Initialiser la population
Evaluer l'adaptation de chaque individu de la population
Tant que ( / Condition de terminaison ) Faire
t := t+1
Sélectionner les meilleurs individus pour la
reproduction
Appliquer les opérateurs génétiques
Evaluer chaque individu de la génération enfant
Fait
Fin

```

**Figure 4.2.** Structure générale d'un algorithme génétique

- *sélection* : détermine, par application d'une méthode probabiliste, les individus jugés adaptés et ce, en vue de les cloner à la génération suivante ;
- *croisement* : consiste à appliquer sur la population enfant, un opérateur de combinaison des caractéristiques, avec une probabilité  $P_c$  donnée ;

– *mutation* : consiste à muter chaque individu issu de la population croisée, avec une probabilité  $P_m$  donnée ;

– *extraction de l'individu solution* : consiste à déterminer l'individu solution, caractérisé par la meilleure valeur d'adaptation, puis à l'interpréter sur la base du modèle défini préalablement.

#### 4.4.1.1. Modélisation génétique

La modélisation d'un processus d'optimisation sous l'angle de la génétique revient à définir, dans le cadre du problème posé, ce qu'est formellement l'individu, la population, la fonction d'adaptation et les opérateurs génétiques.

INDIVIDU ET POPULATION.– Chaque individu ou *chromosome* exprimé par un *génotype*, est constitué d'un ensemble fixe de *gènes* représentant chacune de ses caractéristiques. Le décodage d'un individu produit son *phénotype*. Un gène identifié par sa position appelée *locus*, peut prendre plusieurs valeurs dénommées *allèles* constituant ainsi l'alphabet de l'individu. Initialement, on adopta particulièrement la représentation binaire, ce qui correspond à l'alphabet minimal  $\{0,1\}$  ; on parle alors de *version canonique* des algorithmes génétiques. Par la suite, d'autres représentations étendues ont été présentées. L'efficacité du codage retenu pour la représentation des individus dépend essentiellement du respect des principes de pertinence des briques élémentaires et minimisation de l'alphabet [GOL 94].

FONCTION D'ADAPTATION.– Chaque individu solution a une valeur *fitness* retournée par l'application d'une fonction d'évaluation qui doit être capable de favoriser la sélection d'individus dans la direction de l'optimum qui est à priori inconnue.

OPÉRATEURS GÉNÉTIQUES.– Les opérateurs génétiques représentent des procédures de transformation des individus entre deux générations. Les algorithmes génétiques exploitent principalement trois types d'opérateurs visant chacun d'eux un objectif spécifique relativement à la couverture de l'espace des solutions. Ces opérateurs sont la sélection, le croisement et la mutation :

– le principe de sélection est tel que les individus les mieux adaptés fournissent la descendance la plus élevée. La tendance à converger vers des optima locaux est évitée grâce à des techniques de réduction de la pression sélective [TAL 99] et de nichage [MAH 95, PET 96] ;

– le croisement est le second opérateur génétique appliqué à la population d'individus issus de la sélection. C'est un opérateur de combinaison qui agit généralement par paires en déterminant un ou plusieurs points de coupure, délimitant les frontières des parties à échanger ;

– la mutation consiste à modifier la valeur d'un gène. Des techniques ont été proposées dans le but de limiter la portée d'exploration de la mutation en cours de générations de l'algorithme [FRE 99, MIC 96].

#### 4.4.1.2. *Analyse formelle*

Des travaux sur la formalisation des algorithmes génétiques ont apporté une justification rationnelle du processus d'optimisation induit [CER 94, HAR 91, HOL 75].

A ce titre, le théorème fondamental montre que l'approche génétique de modélisation de problèmes d'optimisation présente deux avantages majeurs. Le premier réside dans la capacité de l'algorithme à traiter simultanément, pour une population de taille  $N$ , un nombre de directions de recherche de l'ordre de  $N^3$ . Ceci traduit la propriété fondamentale des algorithmes génétiques connue sous le qualificatif de parallélisme implicite et qui justifie leur application à des problèmes d'optimisation caractérisés par des espaces complexes [MIC 96]. Le second avantage porte sur la faculté de l'algorithme à résoudre efficacement le dilemme exploration/exploitation. En effet, Holland développe la théorie des schèmes [HOL 75] et montre que l'algorithme attribue un nombre exponentiellement croissant à la meilleure direction observée.

Par ailleurs, Aarts *et al.* [AAR 89] prouvent, en utilisant la théorie des chaînes de Markov, la convergence des algorithmes génétiques sous conditions que les stratégies de reproduction et sélection soient élitistes. Anckenbrandt [ANC 90] confirme ce résultat en fournissant en outre, des estimations du coût de la solution produite.

#### 4.4.2. *Recherche d'information basée sur la génétique*

L'utilisation de techniques génétiques dans le but d'appréhender la problématique en recherche d'information est d'emblée justifiée par leur efficacité intrinsèque à explorer des espaces de recherche complexes. Vus sous l'angle de l'apprentissage, les stratégies de recherche basées sur la génétique ciblent principalement les objectifs suivants [TAM 01] :

- *représentation optimale des documents* : l'idée est d'adapter les descripteurs de documents aux profils des utilisateurs lors d'un épisode unique de recherche, par application des opérateurs génétiques sur des générations de populations de descripteurs de documents [GOR 88] ;

- *représentation optimale des requêtes* : ceci revient à appliquer les techniques de reformulation de requêtes sur une population de requêtes et non une unique requête comme préconisé dans l'approche classique [BOU 99b, KRA 95, TAM 01, YAN 93]. De nombreux travaux ont montré à ce titre, l'intérêt de l'évaluation multi-requêtes [BEL 93, LEE 97] ;

- *modélisation génétique du processus de recherche* : dans ce cas, l'objectif est généralement de diffuser le processus de recherche des documents pertinents sur un

ensemble d'agents de recherche évoluant sur différentes régions documentaires. Les résultats d'évaluation de chaque agent sont alors combinés pour construire la réponse du système [MEN 99, NIC 01].

Pour des raisons pratiques, nous synthétisons dans la suite, les travaux les plus illustratifs du domaine, classés selon les objectifs cités ci-dessus.

#### 4.4.2.1. Traitement génétique des documents

L'originalité de l'approche génétique pour l'optimisation de documents est due à Gordon [GOR 88, GOR 91]. L'auteur propose une méthode adaptative de représentation des documents, dans le modèle probabiliste, basée sur les algorithmes génétiques.

L'algorithme opère sur chaque document en lui associant  $N$  descriptions dont chacune est définie par une liste de termes d'indexation non pondérée. Le renouvellement des générations de descripteurs de documents est basée sur l'utilisation de requêtes d'apprentissage pertinentes et autres non pertinentes. L'algorithme génétique a la structure présentée dans la figure 4.3.

La qualité de chaque descripteur  $Desc\_D_i$  de chaque document  $D$  est évaluée à l'aide de la formule suivante :

$$Fitness(Desc\_D_i) = Score(Desc\_D_i, R\_P) + w * (G_g^{NP} - (Score(Desc\_D_i, R\_NP) - G_g^{NP})) \quad [4.14]$$

où :

$R\_P$  : ensemble des requêtes pertinentes pour le document  $D$

$R\_NP$  : ensemble des requêtes non pertinentes pour le document  $D$

$Score(Desc\_D_i, R\_P)$  : score moyen de ressemblance du descripteur avec les requêtes pertinentes

$Score(Desc\_D_i, R\_NP)$  : score moyen de ressemblance du descripteur avec les requêtes non pertinentes

$G_g^{NP}$  : score de ressemblance moyen de la population de descripteurs, à la génération  $g$ , avec les requêtes non pertinentes

$w$  : constante

$$Score(Desc\_D_i, R\_P) = \frac{1}{M} \sum_{k=1}^M J(Desc\_D_i, q_k^P) \quad [4.15]$$

$$G_j^{NP} = \frac{1}{M * N} \sum_{i=1}^N \sum_{k=1}^M J(Desc\_Di, q_k^{NP}) \quad [4.16]$$

$q_k^P$  : requête pertinente pour  $D$

$q_k^{NP}$  : requête non pertinente pour  $D$

$M$  : nombre de requêtes pertinentes pour  $D$

$N$  : nombre de descripteurs du document  $D$

$J$  : mesure de Jaccard

Début

Générer la population initiale de descripteurs du document  $D$

Répéter

Evaluer chaque descripteur

Remplacer la génération courante de descripteurs en considérant

- la structure des descripteurs courants

- le degré de ressemblance avec les requêtes pertinents et requêtes non pertinentes

Jusqu'à atteindre un critère d'arrêt

Fin

**Figure 4.3.** Algorithme génétique pour l'optimisation de descripteurs de documents [GOR 88]

Des opérateurs classiques de croisement et mutation sont ensuite appliqués à chaque génération de descripteurs. Les expérimentations réalisées sur une base de test locale font état d'un accroissement de performances évalué à 25 % à la 40<sup>e</sup> génération.

L'auteur montre que l'algorithme génétique produit des descriptions de documents plus performantes que celles générées dans le modèle probabiliste. Gordon exploite ces premiers résultats pour définir un mécanisme de classification des documents [GOR 91] basé sur le regroupement de documents pertinents à une même requête. La technique de classification permet essentiellement de regrouper des documents copertinents dans la même classe. L'expérimentation de l'approche sur une base de tes locale révèle que la redescription génétique des classes de documents permet d'atteindre 39,74 % d'accroissement des performances au bout de 20 générations et 56,61 % au bout de 40 générations.

4.4.2.2. *Traitement génétique des requêtes*

Dans ce cadre, plusieurs travaux ont étudié la faisabilité du processus de reformulation de requête par injection de pertinence, en utilisant les principes de la génétique [CHE 95, HOR 00, KRA 95, YAN 93].

```

Début
Evaluer Requête-Utilisateur
t :=0
Construire la population initiale de niches de
requêtes  $Pop^{(0)}$ 
Répéter
Pour chaque niche de requêtes  $N_i \in Pop^{(t)}$ 
Pour chaque individu requête  $\in N_i$ 
Effectuer la recherche
Identifier les nouvelles niches  $N_{nouveau_i}$ 
Effectuer la fusion des listes de documents issues
de l'évaluation des requêtes de chaque niche
Retenir le jugement utilisateur des meilleurs
documents
Pour chaque nouvelle niche  $N_{nouv_i}$ 
Calculer l'adaptation des individus requêtes
Appliquer les opérateurs génétiques
t :=t+1
Jusqu'à arrêt
Fin

```

**Figure 4.4.** *Algorithme génétique pour l'optimisation de requêtes [TAM 03]*

Comparativement à ces travaux, l'approche développée dans [TAM 03], est caractérisée par l'utilisation d'une fonction d'adaptation ajustée par la technique de nichage en vue de résoudre le problème de multimodalité de la pertinence et de l'utilisation d'opérateurs génétiques spécifiques à la reformulation de requêtes car augmentés par une connaissance du domaine.

Le processus général a pour but d'optimiser les descripteurs de requêtes (voir figure 4.4) ; il est fondé sur l'évolution de niches de requêtes. Une niche est un groupe potentiel d'individus, représentés par des requêtes, qui investit une direction de recherche donnée et évolue en accord avec les valeurs d'adaptation des requêtes associées et des jugements de pertinence de l'utilisateur.

---

1. Propriété qui traduit le fait que des documents possédant des descripteurs relativement dissemblables peuvent être associés à une même requête.



A cet effet, on définit l'opérateur *coniche* noté  $\equiv_N$  comme suit :

$$(Q_u^{(s)} \equiv_N Q_v^{(s)}) \Leftrightarrow ((Ds(Q_u^{(s)}, L) \cap Ds(Q_v^{(s)}, L) > Limite\_Coniche) \quad [4.17]$$

où :

$Q_u^{(s)}$  : individu requête à la génération  $s$  de l'algorithme génétique

$Ds(Q, L)$  : ensemble des  $L$  premiers documents sélectionnés par la requête individu  $Q$

$Limite\_Coniche$  : nombre minimal de documents communs retrouvés par les individus requêtes d'une même niche.

avec :

$Limite\_Coniche = NbJug * Prop\_Coniche$

$Prop\_Coniche$  : constante réelle appartenant à l'intervalle  $[0 \ 1]$

$NbJug$  : nombre de documents jugés par l'utilisateur

A chaque individu requête est associé une mesure d'adaptation, notée fitness, définie par la formule :

$$Fitness(Q_u^{(s)}) = 1 + \frac{\sum_{dr \in Dr, dnr \in Dnr} J(Q_u^{(s)}, dr^{(s)}) - J(Q_u^{(s)}, dnr^{(s)})}{\sum_{dr \in Dr, dnr \in Dnr} J(Q_h^{(s)}, dr^{(s)}) - J(Q_u^{(s)}, dnr^{(s)})} \quad [4.18]$$

où :

$Dr$  : ensemble de documents pertinents retrouvés à travers les générations de l'algorithme génétique

$Dnr$  : ensemble de documents non pertinents retrouvés à travers les générations de l'algorithme génétique

$J(D_j, Q_u^{(s)})$  : mesure de Jaccard définie par :

$$J(D_j, Q_u^{(s)}) = \frac{\sum_{i=1}^T q_{ui} d_{ji}}{\sum_{i=1}^T q_{ui}^2 + \sum_{i=1}^T d_{ji}^2 - \sum_{i=1}^T q_{ui}^{(s)}} \quad [4.19]$$

avec :

$q_{ui}^{(s)}$  : poids du terme  $t_i$  dans la requête

$Q_u^{(s)}, d_{ji}$  : poids du terme  $t_i$  dans le document

$D_j, T$  : nombre total de termes d'indexation

Par ailleurs, les opérateurs génétiques ne sont pas classiques. Ils sont définis selon les principes d'expansion et repondération de requêtes dans le but de limiter l'espace de recherche et partant, de faire converger l'algorithme à un nombre réduit de générations de requêtes. L'approche a été validée sur les collections de test TREC [BOU 99b, TAM 03].

Les expérimentations réalisées montrent que l'optimisation génétique assure un accroissement de performances de 18 % à 63 % en fonction des générations et ce, relativement à une reformulation classique basée sur une seule requête.

De plus, il en ressort que l'intégration de la connaissance à la structure des opérateurs génétiques est à l'origine d'un taux d'accroissement des résultats de 5 % à 15 % comparativement à l'application d'opérateurs classiques.

#### 4.4.2.3. Processus génétique de recherche

L'utilisation des techniques génétiques pour la modélisation d'un processus de recherche adaptatif a été essentiellement proposée et validée dans un environnement WEB [MEN 99, NIC 01].

L'approche présentée dans [MEN 99] est basée sur la mise en œuvre d'une population d'agents de recherche qui naviguent à travers le réseau d'information. Ces agents évoluent selon un algorithme génétique, illustré par la figure 4.5, qui optimise la pertinence supposée des documents visités, en réduisant les coûts de recherche.

Les principales étapes de l'algorithme sont les suivantes :

##### 1. Génération de la population initiale

L'utilisateur produit initialement une liste de mots clés et documents pertinents ( $D_1, \dots, D_p$ ). Chaque agent est alors positionné sur un de ces documents avec une énergie initiale  $E_0 = \theta/2$ .

##### 2. Sélection d'un lien de navigation

Un agent est situé sur un document présentant plusieurs liens de référence. Chacun d'eux est caractérisé par les termes les plus proches  $k_1 \dots k_l$ , et modélisé à l'aide d'un réseau de neurones.

La sélection du lien de navigation est effectuée en deux opérations : calcul de sa valeur d'activation puis calcul de sa probabilité de sélection.

```

a : agent
D : document
Ea : énergie de l'agent a
C(D) : coût d'atteindre le document D : durée de
transfert, longueur du document...
E(D) : valeur de pertinence d'un document D
θ : constante

Début
Initialiser une population de N agents avec une
énergie initiale E=θ / 2
Refaire :
Pour chaque agent a
Sélectionner un lien à partir du document courant
Consulter le document référé D
Mettre à jour Ea = Ea - C(D) + E(D)
Apprendre par renforcement de l'utilisateur le
signal e(D)
Si ( Ea ≥ θ ) Alors
    à = Mutation (Croisement (Clone (a)))
    Ea = Ea / 2
Sinon Si ( Ea < 0 )
    Eliminer (a)
Récupérer le jugement de pertinence de l'utilisateur
Refait
Fin

```

**Figure 4.5.** Algorithme génétique de recherche d'information dans un environnement WEB [MEN 99]

#### Calcul de la valeur d'activation des liens

Pour chaque lien de référence  $l$  figurant dans le document courant, on calcule la valeur d'activation propagée  $\lambda_l$  comme suit :

$$\lambda_l = \sum_{j=1}^n (b_j + \sum_{k=1}^m w_{jk} In_k^l) \quad [4.20]$$

où :

$b_j$  : facteur de biais numéro  $j$

$w_{jk}$  : poids de la liaison du nœud lien  $j$  et terme  $k$

$In_k^l$  : valeur d'activation en sortie du lien  $l$  à partir du terme  $k$

$n$  : nombre de termes du lien  $l$

$m$  : nombre total de liens dans le document

avec :

$$In_k^l = \sum_{i / dist(k_i, l) < \rho} \frac{1}{dist(k_i, l)}$$

où :

$dist(k_p, l)$  : nombre de liens où intervient le terme  $k_i$

La distance  $dist(k_p, l) < \rho$  limite le comptage de liens à une fenêtre de  $\rho$  liens.

*Calcul de la probabilité de sélection des liens*

L'agent calcule de manière stochastique un score de sélection  $Pr[l]$  de liens de navigation selon la distribution de probabilité suivante :

$$Pr[l] = \frac{e^{\beta \lambda_l}}{\sum_{l' \in l} e^{\beta \lambda_{l'}}} \quad [4.21]$$

où :

$\beta$  : constante

Le lien de plus grande valeur de  $Pr[l]$  est alors emprunté par l'agent.

#### 4. Calcul de la valeur d'adaptation d'un agent

Suite à la sélection d'un lien de référence, le document atteint est consulté puis on met à jour l'énergie de l'agent qui traduit sa valeur d'adaptation. La mise à jour exprime un gain ou une perte en fonction des jugements de pertinence préalables ; elle est le résultat de l'application de la formule :

$$Ea = Ea - C(D) + E(D) \quad [4.22]$$

où :

$$E(D) = \begin{cases} \chi^* \phi(D) & \text{si } D \text{ est préalablement jugé} \\ \tanh\left(\sum_{k \in D} freq(k, D) * I_k\right) & \text{sin on} \end{cases}$$

avec :

$\chi^*$  : constante

$\phi(D) \in [-1, 0, +1]$  : jugement de pertinence de l'utilisateur

$\tanh$  : fonction tangente hyperbolique  $\in [-1 + 1]$

$freq(k,D)$  : fréquence du terme  $k$  dans le document  $D$ , normalisée par la longueur du document

$I_k$  : facteur de pertinence du terme  $k$ , variable en cours d'évolution de l'algorithme selon la formule :

$$I_k = \alpha * I_k + (1 - \alpha) * w_k * (1 + \log(\frac{1}{C_k}))$$

avec :

$\alpha$  : terme d'inertie

$w_k$  : valeur de pertinence cumulée du terme  $k$  calculé par la formule  $w_k = w_k + \phi(D)$

$C_k$  : proportion de documents pertinents sauvegardés et contenant le terme  $k$

##### 5. Apprentissage du réseau de liens de référence

Un signal de renforcement est calculé selon la formule :

$$\delta(D) = E(D) + \mu \text{Max}_{l \in D} \{\lambda\} - \lambda D \quad [4.23]$$

où :

$\mu$  : constante

Le poids des liens de référence sont alors corrigés par application de l'algorithme de rétropropagation.

##### 6. Application des opérateurs génétiques

L'évolution génétique de chaque agent est basée sur la valeur de son énergie ; il est reproduit dans le cas où son énergie est positive, éliminé dans le cas contraire. Les deux opérateurs de croisement et mutation sont appliqués aux agents.

Deux types de croisement sont définis :

– croisement local : un agent n'est combiné qu'avec un agent situé sur le même document,

– croisement global : un agent peut être combiné avec tout autre agent.

La sortie de l'algorithme est un flot de liens de référence ordonnés par les valeurs de pertinence estimées en partie en fonction du jugement de pertinence de l'utilisateur. L'algorithme s'arrête au vœu de l'utilisateur ou par absence de liens pertinents.

L'approche a été évaluée sur une partie de la collection *Human Society*, contenant 19 427 documents organisés en hypergraphe. Les expérimentations réalisées montrent globalement l'intérêt de l'approche.

Plus précisément, les résultats montrent l'avantage de combiner la capacité de recherche d'information locale des agents et capacité de recherche d'information globale et hétérogène due à l'application de la génétique d'une part et d'une vision de pertinence plus large de l'utilisateur, d'autre part.

#### **4.4.3. Synthèse**

Il ressort principalement de cette brève étude que les algorithmes génétiques trouvent un champ d'application porteur dans le domaine de la recherche d'information.

D'un point de vue large, les travaux montrent que leur propriété de parallélisme implicite permet de réduire la complexité d'exploration d'un espace documentaire. Ces algorithmes supportent des stratégies de recherche permettant de déterminer des critères de choix pour la redescription de requêtes et documents ou redéfinition de la mesure de pertinence dans un contexte d'apprentissage.

La modélisation génétique mérite cependant une analyse conséquente portant sur :

- la valorisation des nombreux paramètres associés (taille de la population, probabilités d'application des opérateurs, stratégie de sélection, etc.),
- l'estimation du coût de convergence,
- et la stabilité de l'algorithme face à la variabilité des types et volumes des collections interrogées.

De la qualité de cette analyse, découle en partie la qualité des performances globales de l'algorithme.

#### **4.5. Conclusion**

Le domaine de la recherche d'information n'est pas isolé de l'évolution des autres domaines de l'informatique.

Dans ce chapitre, nous avons particulièrement examiné les principes du modèle connexioniste et des algorithmes génétiques et avons montré comment ils peuvent être abordés pour répondre à des aspects problématiques posés par la sélection d'informations pertinentes. Nous nous sommes volontairement orientés vers

l'utilisation de telles techniques dans le contexte de systèmes adaptatifs dont le processus d'apprentissage est basé sur les interactions avec les utilisateurs.

Dans un premier volet, il en ressort essentiellement que la théorie du connexionisme constitue, à part entière, un support de modélisation des processus de représentation, sélection de l'information et apprentissage. Les travaux y afférents diffèrent principalement par la structure du réseau et algorithmes mis en œuvre pour l'évaluation et correction de la mesure de pertinence. Selon les choix faits pour ces paramètres, les propriétés du modèle seront fort diverses.

Dans un second volet, on montre que les algorithmes génétiques sont d'avantage utilisés pour élaborer des stratégies plutôt que des modèles de recherche d'information. Exploitant leur efficacité théoriquement prouvée à explorer des espaces de recherche complexes, les travaux d'application en recherche d'information utilisent les techniques génétiques pour développer des approches novatrices de reformulation de requêtes, évaluation et réinjection de pertinence, prédiction des profils utilisateurs, etc. De nombreux paramètres de l'algorithme sont déterminants pour son efficacité. Leur formalisation est difficile et nécessite généralement un ajustement expérimental préalable.

Un effort de réflexion sur la formalisation des relations de corrélation entre d'une part, les caractéristiques des modèles connexistes et algorithmes génétiques et d'autre part, les propriétés du référentiel de la recherche d'information (structure de l'information, volume, type d'utilisation, etc.) serait sans doute opportun et ce, dans le but de mettre en œuvre des systèmes de recherche d'information flexibles utilisant les éléments de ces théories.

#### 4.6. Bibliographie

- [ANK 90] ANKENBRANDT C., « An extension to the theory of convergence and a proof of the time complexity of genetic algorithms », *FOGA90*, p. 53-58, 1990.
- [AAR 89] AARTS E.H.L., EIBEN A.E., VANHEE K.M., « A general theory of genetic algorithms », Computing Science Notes, Eindhoven University of Technology Netherlands, 1989.
- [BAE 91] BAECK T., HOFFMEISTER F., SCHWEFEL H.P., « A survey of evolution strategies », *International Conference on Genetic Algorithms*, p. 2-9, 1991.
- [BAR 94] BARRY C.L., « User-defined relevance criteria : an exploratory study », *Journal of the American Society for Information Science*, 45(3), p. 149-159, 1994.
- [BEL 89] BELEW R., « Adaptive information retrieval », *Proceedings of the twelfth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, p. 11-20, 1989.

- [BEL 93] BELKIN N.J., COOL C., BRUCE CROFT W., CALLAN J.P., « Effect of multiple query representations on information retrieval system performance », *Proceedings of ACM SIGIR, Conference on Research and Development in Information Retrieval*, p. 339-346, Pittsburgh, 1993.
- [BOU 92] BOUGHANEM M., Systèmes de recherche d'information : d'un modèle classique à un modèle connexioniste, Thèse de l'Université Paul Sabatier de Toulouse, 1992.
- [BOU 97] BOUGHANEM M., « Mercure at TREC 6 », Harman D.K. (dir.), *sixth International Conference on Text Retrieval TREC 6*, NIST SP, p. 321-328, 1997.
- [BOU 99a] BOUGHANEM M., SOULE-DUPUY C., « Query modification based on relevance backpropagation in adhoc environment », *Information Processing and Management*, 35(2), 1999.
- [BOU 99b] BOUGHANEM M., CHRISMENT C., TAMINE L., « Genetic Approach to Query Space Exploration. Information Retrieval », *Journal of Information Retrieval*, 1(3) p. 175-192, 1999.
- [BRA 94] BRAJNIK G., TASSO C., « A shell for developing non-monotonic user modelling systems, *Journal of human computer studies*, p. 31-62, 1994.
- [BRU 94] BRUCE H.B., « A cognitive view of the situational dynamism of user-centred relevance estimation », *Journal of the American Society for Information Science*, 45(3), p. 142-148, 1994.
- [BUC 94] BUCKLEY C., SALTON G., ALLAN J., « The effect of adding information in a relevance feedback environment », *Conference on Research and Development in Information Retrieval (SIGIR)*, p. 292-300, 1994.
- [CER 94] CERF R., Une Théorie asymptotique des algorithmes génétiques, Thèse Phd, Université de Montpellier II, mars 1994.
- [CHE 95] CHEN H., « Machine learning for information retrieval: Neural networks, symbolic Learning and genetic Algorithms », *Journal of American Society on Information System*, 46(3), p. 194-216, 1995.
- [CRE 94] CRESTAN F., « Comparing neural and probabilistic relevance feedback in an interactive information retrieval system », *Proceedings of the IEEE International Conference on Neural Networks*, p. 3226-3230, 1994.
- [CRO 79] CROFT W., HAPER D., « Using probabilistic models for document retrieval without relevance information », *Journal of Documentation*, p. 185-202, 1979.
- [CRO 94] CROUCH C., CROUCH D., NAREDDY, K., « Associative and adaptive retrieval in a conectionist system », *International Journal of Expert Systems*, 7(2), p. 193-202, 1994.
- [DEA 99] DEAN MONIKA J., HENZINGER R., « Finding Related Pages in the World Wide Web », *Proceedings of the Eighth World-Wide Web Conference*, Toronto, Canada, p. 1467-1479, 1999.



- [DEN 97] DENOS N., Modélisation de la pertinence en recherche d'information : modèle conceptuel, formalisation et application, Thèse de doctorat, Joseph Fourier Grenoble 1, 1997.
- [FRE 99] FREITAS A.A., « A genetic algorithm for generalized rule induction », *R. Roy et al. advances in soft computing engineering design & manufacturing In Proceedings of the third on line world conference on soft computing*, Springer Verlag, p. 340-353, 1999.
- [GOL 89] GOLDBERG D.E., *Genetic algorithms in search, optimisation and machine learning*, Addison Wesley, 1989.
- [GOL 94] GOLDBERG D.E., *Algorithmes génétiques, exploration, optimisation et apprentissage automatique*, Addison Wesley, 1994.
- [GOR 88] GORDON M., « Probabilistic and Genetic Algorithms for Document Retrieval », *Communications of the ACM*, p. 1208-1218, octobre 1988.
- [GOR 91] GORDON M.D., « User-based document clustering by redescribing subject descriptions with a genetic algorithm », *Journal of The American Society for Information Science*, 42(5) p. 311-322, 1991.
- [GRE 95] GREEN R., « Topical relevance relationships. I. Why topic matching fails », *Journal of the American Sociociety for Information Science*, 46(9), p. 646-653, 1995.
- [HAR 91] HARTMAN W., BELEW R.K., « Optimizing an arbitrary function is hard for the genetic algorithm », *Proceedings of the International Conference on Genetic Algorithms* p. 190-195, 1991.
- [HEB 49] HEBB D.O., *The Organization of Behaviour*, J. Wiley & Sons, New York, 1949.
- [HOL 75] HOLALND J., « Adaptation in Natural and Artificial Systems », University of Michigan Press, Ann Arbor, 1975 .
- [HOR 00] HORNG J.T., YEH C.C., « Applying genetic algorithms to query optimisation in information retrieval », *Information Processing and Management*, 36(5), p. 737-759, 2000.
- [HOW 94] HOWARD D.L., « Pertinence as refelcted in personal constructs », *Journal of the American Sociociety for Information Science*, 45(3), p. 172-185, 1994.
- [KLE 98] KLEINBERG J.M., « Uthoritative sources in a hyperlinked environment », *Proceedings of ACM-SIAM Symposium on Discrete Algorithms*, p. 668-677, 1998.
- [KOZ 92] KOZA J.R., *Genetic programming*, Bradford book, MIT Press, Cambrige, MA, 1992.
- [KRA 95] KRAFT D.H., PETRY F.E., BUCKLES B.P, SADISAVAN T., « Applying genetic algorithms to information retrieval system via relevance feedback », Bosc et Kacprzyk (dir.), *Fuzziness in Databse Management Systems Studies in Fuzziness Series*, Physica Verlag, Heidelberg, Allemagne, p. 330-344, 1995.

- [KWO 89] KWOK K.L., « A neural network for probabilistic information retrieval », *Proceedings of ACM SIGIR, Conference on Research and development in Information Retrieval*, p. 21-30, 1989.
- [KWO 99] KWOK K.L., GRUNFELD L., CHAN M., « TREC-8 Adhoc, query and filtering track using PIRCS », *Proceedings of TREC-8*, 1999.
- [LEC 86] LE CUN Y., « Learning process in an assymmetric threshold network », E. Bienenstock *et al.* (dir.), *Disordered systems and biological organizations*, NATO-OASI Series Springer Verlag, 1986.
- [LEE 97] LEE J.H., « Analyse of multiple evidence combination », *Proceedings of ACM SIGIR, Conference on Research and development in Information Retrieval*, p. 267-275, 1997.
- [MCL 86] MCCLELLAND J., RUMELHART D., PDP Research Group, *Parallel distributed processing*, Academic press, New-York, 1986
- [MAH 95] MAHFOUD S.W., Niching methods for genetic algorithms, PHD Thesis, University of Illinois at Urbana, Champaign, 1995.
- [MEN 99] MENCZER F., BELEW R.K., « Adaptive Retrieval Agents: Internalizing Local Context and Scaling up to the WEB », *Machine Learning*, p. 1-45, Kluwer Academic Publishers, 1999.
- [MIC 96] MICHALEWICZ Z., *Genetic Algorithms + Data Structures = Evolutionary Programs*, Springer Verlag, New York, 3<sup>e</sup> édition, 1996.
- [NIC 01] NICK Z.N., THEMIS P., « Web search using a genetic algorithm », *IEEE Internet Computing Journal*, 5(2) p. 18- 26, 2001.
- [PET 96] PETROWSKI A., « A clearing procedure as a niching method for genetic algorithms », *Proceedings of the IEEE International Conference on Evolutionary Computation ICEC*, Nagoya, Japon, p. 798-803, 1996.
- [RIC 83] RICH E., « Users are individuals: individualising user models », *Journal of Man Machine Studies*, 18(1), p. 199-214, 1983.
- [ROB 94] ROBERTSON S., WALKER S., « Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval », *Proceedings of the seventh Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, p. 232-241, 1994.
- [ROC 71] ROCCHIO J.J., « Relevance feedback in information retrieval », G. Salton (dir.), *The Smart System Experiments in Automatic Document Processing*, Prentice-Hall, Englewood Cliffs, NJ, p. 313-23, 1971.
- [SAL 89] SALTON G., *Automatic Text Processing. The Transformation Analysis and Retrieval of Information by Computer*, Addison Wesley, Reading, 1989.
- [TAL 99] TALBI G., Métaheuristiques pour l'optimisation combinatoire multi-objectifs : état de l'art, Rapport CENT France Télécom, 1999.

- [TAM 01] TAMINE L., BOUGHANEM M., « Un algorithme génétique spécifique à une reformulation multi-requêtes dans un système de recherche d'information », *Revue IS en Sciences et Traitement de l'Information*, 1(1) p. 49-76, 2001.
- [TAM 03] TAMINE L., CHRISMENT C., BOUGHANEM M., « Multiple query evaluation based on an enhanced genetic algorithm », *Information Processing and Management*, 39(2), p. 215-231, 2003.
- [YAN 93] YANG J.J., KORFHAGE R.R., « Query optimisation in information retrieval using genetic algorithms », *International Conference on Genetic Algorithms*, p. 603-613, 1993.