Raisonnement à base de cas textuels – état de l'art et perspectives

Luc Lamontagne – Guy Lapalme

Université de Montréal,
Département d'informatique et de recherche opérationnelle (DIRO)
CP 6128, Succ. Centre-ville, Montréal (Québec)
Canada H3C 3J7
{lamontal,lapalme}@iro.umontreal.ca

RÉSUMÉ. *Traditionnellement le raisonnement à base de cas (CBR) s'appuie sur des expériences décrites dans des formats complètement structurés tels que des objets ou des enregistrements de base de données. Toutefois d'autres modèles ont été proposés pour surmonter les limitations de cette approche structurelle et rendre possible l'application à des domaines plus variés. Dans cet article, nous passons en revue les extensions du formalisme CBR proposées pour traiter des expériences décrites dans des documents textuels, travaux regroupés sous la bannière CBR textuel. Après une présentation succincte des principes généraux du raisonnement à base de cas, nous décrivons les principaux travaux du CBR textuel et nous les comparons selon différents aspects techniques et applicatifs. Finalement, nous proposons quelques problèmes et avenues de recherche méritant d'être explorés dans des travaux futurs.*

ABSTRACT. *Traditionnally case-based reasoning is conducted on experiences that are represented in a well structured format such as objects or database records. However different models have been proposed to overcome some of the limitations imposed by this structural approach and to undertake new applications domains. In this paper, we review some of the extensions proposed to apply CBR principles to experiences contained in textual documents, commonly refered to as textual CBR. Following a short presentation of some case-based reasoning principles, we present the main research works in Textual CBR and provide a synthesis of current state of the art in the field. We finally discuss some shortcomings of the current approaches and propose some directions for future research.*

MOTS-CLÉS : *raisonnement à base de cas, cas textuels, adaptation, recherche d'information.*

KEYWORDS : *case-based reasoning, textual cases, adaptation, information retrieval.*

1. Introduction

Traditionnellement le raisonnement à base de cas (CBR) s'appuie sur des expériences décrites dans des formats complètement structurés tels que des objets ou des enregistrements de base de données. Ce formalisme a permis au CBR de prendre un essor important au cours de la dernière décennie grâce, entre autre, à de nombreuses applications commerciales qui se sont avérées fructueuses. Toutefois les praticiens du domaine ont rapidement constaté les limites de cette approche structurelle et ont proposé d'autres modèles pour en surmonter les difficultés et étendre son application à des domaines plus variés.

Nous nous intéressons plus particulièrement aux extensions du formalisme CBR pour traiter des expériences décrites dans des documents textuels, travaux regroupés sous la bannière CBR textuel. Ces approches sont relativement récentes et s'appuient principalement sur des techniques de la recherche d'information, de l'apprentissage automatique et du traitement automatique des langues. La voie du CBR textuel s'avère nécessaire pour certaines applications, telles que celles du domaine de la jurisprudence légale ou du diagnostic médical, dont le raisonnement s'appuie sur des comptes-rendus textuels. Ces travaux sont également motivés par l'avènement des technologies web et l'émergence de pratiques de gestion de connaissance au sein des entreprises favorisant la préservation et l'exploitation d'expériences corporatives.

Dans cet article, nous présentons l'état actuel des travaux de ce domaine. Nous débutons par une présentation succincte des principes généraux du CBR et des différentes familles de modèles de système CBR. Par la suite, nous décrivons les principaux travaux du CBR textuel et nous établissons un tableau comparatif de ces approches. Finalement, nous proposons quelques problèmes et voies de recherche pour des travaux futurs.

2. Principes généraux du raisonnement à base de cas

Le raisonnement à base de cas (CBR) est une approche de résolution de problèmes qui utilise des expériences passées pour résoudre de nouveaux problèmes [LEA 96]. L'ensemble des expériences forme une base de cas. Typiquement un cas contient au moins deux parties : une description de situation représentant un "problème" et une "solution" utilisée pour remédier à cette situation. Parfois, le cas décrit également les conséquences résultant de l'application de la solution (e.g. succès ou échec). Les techniques CBR permettent de produire de nouvelles solutions en extrapolant sur les situations similaires au problème à résoudre. Cette approche est adéquate pour les domaines où la similarité entre les descriptions de problèmes nous donne une indication de l'utilité des solutions antécédentes.

Les fondements du CBR proviennent de travaux en sciences cognitives menés par Roger Schank et son équipe de recherche durant les années 80 [RIE 89]. Leurs

travaux ont mené à la théorie de la mémoire dynamique selon laquelle les processus cognitifs de compréhension, de mémorisation et d'apprentissage utilisent une même structure de mémoire. Cette structure, les "memory organization packets" (MOP), est représentée à l'aide de schémas de représentation de connaissance tels que des graphes conceptuels et des scripts.

Au début de la dernière décennie, on a assisté à un regain de popularité du domaine et de nouvelles tendances qui misent sur la simplification de la représentation des cas et sur des applications à plus grande échelle. Le CBR se révèle alors une précieuse technique pour la mise en œuvre d'applications commerciales [WAT 98] pour différentes tâches telles que la résolution de problèmes (e.g. diagnostic, planification, design), les systèmes d'aide à la décision, les "help desk" et la gestion de connaissances. Ceci en fait l'une des techniques de l'intelligence artificielle les plus largement répandues actuellement.

L'approche CBR offre de nombreux avantages. Pour certaines applications, la démarche CBR est plus simple à mettre en œuvre que celles basées sur un modèle du domaine (e.g. base de règles) ; elle permet d'éviter les problèmes d'acquisition de connaissance ("knowledge bottleneck") qui rendent difficile la conception de bases de connaissances de taille importante. Le CBR est particulièrement bien adapté aux applications dont la tâche est accomplie par des humains expérimentés dans leur domaine et dont les expériences sont disponibles dans une base de données, dans des documents ou chez un expert humain. On l'utilise pour les domaines n'exigeant pas de solution optimale et dont les principes sont mal formalisés ou peu éprouvés.

2.1 Composantes d'un système à base de cas

Un système CBR est une combinaison de processus et de connaissances ("knowledge containers") qui permettent de préserver et d'exploiter les expériences passées. Pour simplifier notre présentation, nous nous appuyons sur le modèle générique présenté dans la Figure 1. On y note comme principaux processus[1] la recherche ("retrieval"), l'adaptation ("reuse"), la maintenance ("retain") et la

[1] Aamodt et Plaza [Aam94] ont proposé un modèle qui représente le raisonnement à base de cas comme un cycle comportant 4 processus: recherche-réutilisation-révision-rétention. Pour simplifier notre discussion, nous intégrons la phase de rétention, qui consiste à intégrer une nouvelle paire problème-solution dans la base de cas, dans la politique de maintenance du système. Une autre phase du modèle, la révision de solution, n'est pas discutée dans ce document. Cette activité consiste à vérifier la validité d'une solution soit par consultation avec l'usager du système, par simulation ou par évaluation numérique.

construction ("authoring")[2] et comme structures de connaissances le vocabulaire d'indexation, la base de cas, les métriques de similarité et les connaissances d'adaptation.

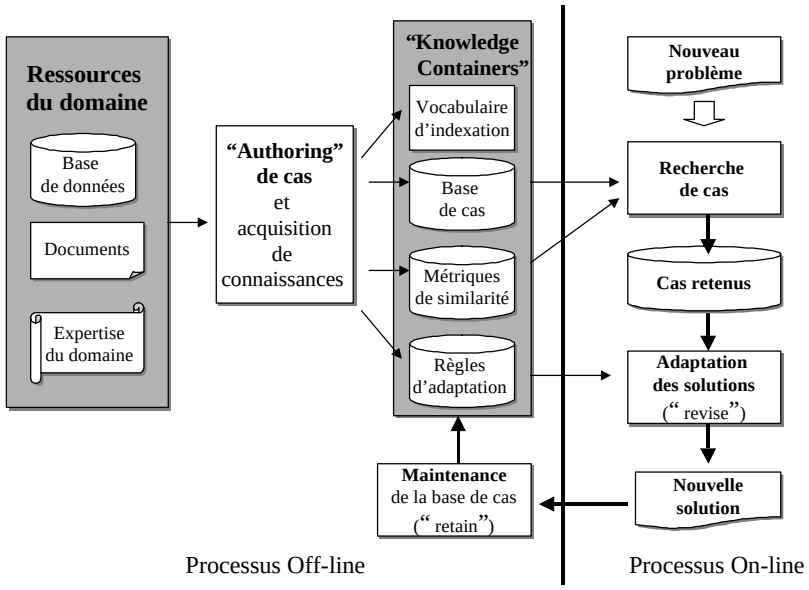

Figure 1. *Modèle générique d'un système CBR*

2.1.1 Processus

La recherche : cette phase permet de déterminer les cas de la base qui sont les plus similaires au problème à résoudre. La procédure de recherche est habituellement implantée par une sélection des plus proches voisins ("k-nearest-neighbors") ou par la construction d'une structure de partitionnement obtenue par induction. L'approche des plus proches voisins utilise des métriques de similarité pour mesurer la correspondance entre chaque cas et le nouveau problème à résoudre. L'approche par induction génère un arbre qui répartit les cas selon différents attributs et qui permet de guider le processus de recherche.

L'adaptation : suite à la sélection de cas lors de la phase de recherche, le système CBR aide l'usager à modifier et à réutiliser les solutions de ces cas pour résoudre son problème courant. En général, on retrouve deux approches pour

[2] La littérature CBR francophone ne propose pas de terme permettant de bien capturer la notion de "authoring". Nous retenons pour cet article le terme "construction". Toutefois d'autres appellations telles que "édition", "ingénieurie", "acquisition"ou "rédaction" nous ont également été proposées.

l'adaptation de cas (figure 2). Par l'approche transformationnelle (ou structurelle), on obtient une nouvelle solution en modifiant des solutions antécédentes et en les réorientant afin de satisfaire le nouveau problème. Par l'approche générative (ou dérivationnelle), on garde, pour chaque cas passé, une trace des étapes qui ont permis de générer la solution. Pour un nouveau problème, une nouvelle solution est générée en appliquant l'une de ces suites d'étapes. Certains travaux visent également à unifier ces différentes approches d'adaptation (voir [FUC 99] et [FUC 00] pour une proposition de modèle général).

Peu de systèmes CBR font de l'adaptation complètement automatique. Pour la plupart des systèmes, une intervention humaine est nécessaire pour générer partiellement ou complètement une solution à partir d'exemples. Le degré d'intervention humaine dépend des bénéfices en terme de qualité de solution que peut apporter l'automatisation de la phase d'adaptation.

Maintenance : durant le cycle de vie d'un système CBR, les concepteurs doivent préconiser certaines stratégies pour intégrer de nouvelles solutions dans la base de cas et pour modifier les structures du système CBR pour en optimiser les performances. Une stratégie simple est d'insérer tout nouveau cas dans la base. Mais d'autres stratégies visent à apporter des modifications à la structuration de la base de cas (e.g. indexation) pour en faciliter l'exploitation. On peut également altérer les cas en modifiant leurs attributs et leur importance relative. Cet aspect de recherche est actuellement l'un des plus actifs du domaine CBR [Lea 01].

Construction : ce processus, en amont des activités de résolution de problèmes du système CBR, soutend la structuration initiale de la base de cas et des autres connaissances du système à partir de différentes ressources tels des documents, bases de données ou transcriptions d'interviews avec des praticiens du domaine. Ce processus, souvent effectué manuellement par le concepteur du système, se prête moins bien à l'automatisation car il nécessite une connaissance du cadre applicatif pour guider, entre autre, la sélection du vocabulaire d'indexation et la définition des métriques de similarités.

2.1.3 Connaissances

Les différentes connaissances utilisées par un système CBR sont regroupées en quatre catégories ("knowledge containers") :

- *vocabulaire d'indexation* : un ensemble d'attributs ou de traits ("features") qui caractérisent la description de problèmes et de solutions du domaine. Ces attributs sont utilisés pour construire la base de cas et jouent un rôle important lors de la phase de recherche.

- *base de cas* : l'ensemble des expériences structurées qui seront exploitées par les phases de recherche, d'adaptation et de maintenance.

- *mesures de similarité* : des fonctions pour évaluer la similarité entre deux ou plusieurs cas. Ces mesures sont définies en fonction des traits et sont utilisées pour la recherche dans la base de cas.

- *connaissances d'adaptation :* des heuristiques du domaine, habituellement sous forme de règles, permettant de modifier les solutions et d'évaluer leur applicabilité à de nouvelles situations.

3. Modèles CBR

Il existe plusieurs modèles pour le raisonnement à base de cas. Ces modèles sont regroupés en trois grandes familles : structurelle, conversationnelle et textuelle. Avant de présenter plus en détail les travaux du CBR textuel, nous décrivons dans les sections suivantes les principales différences entre ces trois familles.

3.1 Modèle structurel

Le modèle structurel a émergé des premières vagues applicatives de systèmes CBR. Dans ce modèle, toutes les caractéristiques importantes pour décrire un cas sont déterminées à l'avance par le concepteur du système. Ainsi, le concepteur élabore un modèle de données du domaine applicatif. Tel qu'illustré à la figure 2, les cas sont complètement structurés et sont représentés par des paires <attribut, valeur> (similaire à un "frame" ou à un objet). D'un point de vue applicatif, un attribut représente une caractéristique importante du domaine d'application. Les échelles de valeurs les plus fréquemment utilisées pour structurer les attributs sont les entiers/réels, les booléens et les symboles. La représentation des cas peut être sur un seul niveau ou sur plusieurs niveaux (hiérarchie d'attributs).

Cas :	*2735*
Entreprise :	*BCE*
Date :	*22/01/2002*
Dividende annuel :	*1,20*
Haut_52_semaines :	*43.70*
Bas_52_semaines :	*32.25*
Dernier_cours :	*35,65*
Recommandation :	*achat*

Figure 2. *Exemple de structuration d'un cas en CBR structurel*

La similarité entre deux cas est mesurée en fonction de la distance entre les valeurs de mêmes attributs. Cette distance est fréquemment estimée par les mesures euclidienne et de Hamming. La similarité globale entre deux cas est habituellement évaluée par une somme pondérée de la similarité de chacun des

attributs. Comme les attributs d'un cas n'ont pas tous la même importance et que cette importance varie d'une situation à l'autre, un poids est attribué à chaque attribut de chaque cas. Ces poids permettent de pondérer la similarité globale entre deux cas en accordant un "vote" plus important aux attributs les plus méritants.

Tous les travaux sur l'adaptation de cas sont menés dans le cadre du modèle structurel. L'adaptation peut varier d'une simple substitution de la valeur d'un attribut jusqu'à la restructuration complète d'une solution. Leake [LEA 96] identifie environ dix techniques permettant de générer des solutions par substitution, transformation partielle ou dérivation complète. Ces techniques sont habituellement mises en œuvre par des systèmes à base de règles, ce qui nous ramène aux problèmes d'acquisition de connaissance et d'absence de principes généraux pour certains domaines. Pour en limiter les difficultés, certaines approches évitent l'adaptation en sélectionnant, durant la phase de recherche, des cas qui nécessiteront peu d'adaptation [SMY 95].

3.2 Modèle conversationnel

Dans l'approche traditionnelle (le modèle structurel), un problème doit être complètement décrit avant que ne débute la recherche dans la base de cas. Cette exigence présuppose une expertise du domaine d'application permettant de bien caractériser une situation à l'aide de valeurs numériques ou symboliques de sélectionner les principaux facteurs pouvant influencer la résolution de son problème. Toutefois pour certains domaines comme le service à la clientèle, ces aspects sont difficiles à déterminer à l'avance, surtout pour les usagers novices de systèmes CBR. Le modèle conversationnel a donc été proposé par Inference Corporation pour surmonter ces difficultés. Il est actuellement le modèle le plus répandu parmi les applications commerciales du CBR.

Comme son nom l'indique, le modèle CBR conversationnel mise sur l'interaction entre l'usager et le système (d'où la notion de "conversation") pour définir progressivement le problème à résoudre et pour sélectionner les solutions les plus appropriées [AHA 01]. Un cas conversationnel consiste en trois parties (voir Figure 3) :

- un problème P : une brève description textuelle, habituellement de quelques lignes, de la nature du problème exprimée.
- une série de questions et de réponses Q_A : des index, exprimés sous forme de questions, permettant d'obtenir plus d'information sur la description du problème. Chaque question a un poids représentant son importance par rapport au cas.
- une action A : une description textuelle de la solution à mettre en œuvre pour ce problème. Cette description n'est pas structurée ("free-text").

Cas : *241*
Titre : *cartouche d'encre endommagée causant des traces noires*
Description : *l'imprimante laisse de petits points noirs sur les deux côtés de la page. Parfois des larges tâches couvrent également la région à imprimer.*
Questions :
Est-ce que les copies sont de mauvaise qualité ? Réponse : oui Score : (-)
Quels types de problèmes avez-vous ? Réponse : trace noires Score : (default)
Est-ce qu'un nettoyage de l'imprimante règle le problème ? Rép : non …
Actions : vérifier la cartouche d'encre et la remplacer si le niveau d'encre est faible

Figure 3. Exemple de cas pour le modèle conversationnel

Cette représentation de cas est donc une extension du modèle structurel avec des attributs de trois types bien précis : description, questions, actions. La notion de trait est étendue à la notion de question afin de pouvoir interroger l'usager.

Dans le schéma de résolution du CBR conversationnel, l'interaction entre le système et l'usager se fait comme suit :

- l'usager fournit au système une brève description textuelle du problème à résoudre et le système calcule la similarité entre cette description et la section "problème" des cas. Le système propose alors à l'usager une série de questions.
- l'usager choisit les questions auxquelles il souhaite répondre. Pour chaque réponse fournie par l'usager, le système réévalue la similarité de chacun des cas. Les questions n'ayant pas reçu de réponse sont présentées par ordre décroissant de priorité.
- lorsqu'un des cas atteint un niveau de similarité suffisamment élevé (i.e. qu'il franchit un seuil), le système propose ce cas comme solution. Si aucun cas n'atteint un degré de similarité suffisant et que le système n'a plus de questions à poser à l'usager, le problème est stocké comme étant non-résolu.

Les systèmes CBR conversationnels n'effectuent pas d'adaptation des solutions passées. Une des raisons est que la portion 'solutions' des cas n'est pas structurée ("free-text"), ce qui rend difficile la formulation de connaissances d'adaptation. Également, il semble que, pour les applications de type "help-desk", les solutions sont relativement faciles à modifier, même par un préposé inexpérimenté. De plus, l'investissement en temps et en efforts consacrés à développer un système d'inférence qui modifie les solutions est difficile à justifier dans ce contexte opérationnel.

3.3 Modèle textuel

Les travaux sur le raisonnement à base de cas textuels portent sur la résolution de problème à partir d'expériences dont la description est contenue dans des documents textuels. Dans cette approche, les cas textuels sont soit non-structurés ou semi-structurés. Ils sont non-structurés si leur description est complètement en "free-text". Ils sont semi-structurés lorsque le texte est découpé en plusieurs portions étiquetées par des descripteurs tels que "problème", "solution", etc. Un cas textuel non-structuré est un cas dont le seul attribut est textuel tandis qu'un cas textuel semi-structuré est un cas dont un sous-ensemble de ses attributs est textuel.

Pour ce modèle, la représentation textuelle des cas joue habituellement un rôle important dans la résolution du problème. Elle peut être une finalité en soi : par exemple, obtenir le texte d'un jugement légal servant de jurisprudence à une nouvelle cause. Elle peut aussi décrire une situation et une solution qui ne peuvent être facilement codifiées selon un schéma de représentation de connaissance.

Cette voie de recherche est relativement récente car les premiers travaux datent du milieu des années 90. A ce jour, aucune représentation standard ne s'est dégagée pour le modèle textuel. Les approches actuelles misent leurs efforts principalement sur la phase de recherche sur la base de cas et ne proposent pas d'avenue pour l'adaptation de solutions textuelles.

Nous pouvons identifier deux pôles importants dans les différents travaux en CBR textuel :

- structuration de cas textuels : on représente les textes selon un nombre limité de traits basés sur des caractéristiques du domaine (concepts, catégories, sujets, mots-clé, etc.). Pour ce pôle de recherche, on vise à structurer le mieux possible les cas textuels afin de tirer profit de techniques développées pour les systèmes CBR structurel. Les efforts sont déployés pour enrichir l'indexation des textes à l'aide de traitements relativement élaborés comme la catégorisation de texte. Cette approche est intéressante pour les applications dont le domaine est restreint. Le projet SMILE [BRU 97] présenté à la section 4.5 en est un exemple.
- extension du modèle de recherche d'information : dans ce pôle de recherche, on élabore des mécanismes de recherche plus sophistiqués tout en gardant le processus d'indexation le plus simple possible. Dans ce cadre, le choix des traits de cas est déterminé à partir de la fréquence de mots-clés ou de syntagmes de référence ("keyphrases"). Les particularités de l'application se reflètent au niveau de la recherche, soit par la définition de mesures de similarité sémantique ou par des extensions au modèle vectoriel de recherche d'information [SAL 83]. Cette approche semble plutôt valide pour les applications génériques qui veulent conserver une indépendance par rapport au domaine d'application. Le projet FAQFinder [BUR 95] présenté à la section 4.1 en est un exemple.

Ces deux pôles sont en fait des stéréotypes auxquels empruntent la plupart des approches actuelles. Nous présentons à la section 4 divers travaux qui illustrent l'exploitation de connaissances du domaine et le contenu linguistique des textes.

Le CBR textuel diffère de l'approche structurelle dans laquelle les textes sont tout simplement des chaînes de caractères sans syntaxe ni sémantique. De plus, cette dernière impose une structuration complète des attributs d'un cas. Nous considérons également que le modèle conversationnel, présenté à la section précédente, ne fait pas partie des approches textuelles. La phase préliminaire du CBR conversationnel se limite à une comparaison, par mots-clé ou n-grammes[3] de caractères, de courtes descriptions textuelles de problèmes. Durant la phase suivante, l'interaction avec l'usager est guidée par une suite de questions et de réponses. Les échanges lors de l'interaction ne font l'objet d'aucun traitement textuel. La langue y est utilisée uniquement dans le but de rendre les questions plus intelligibles à l'usager du système.

4. Principaux travaux en CBR textuel

Dans cette section, nous présentons les travaux que nous jugeons les plus représentatifs de l'état d'avancement du CBR textuel. Ces travaux ont été sélectionnés parce qu'ils apportent des contributions au raisonnement à base de cas et, pour la plupart, ont une influence sur les travaux actuels de la communauté CBR. Ce tour d'horizon donne un aperçu de l'étendu du niveau de structuration des cas, de la complexité des métriques de similarité et des mécanismes de recherche sur la base de cas. Toutefois on retrouve également d'autres travaux combinant CBR et documents textuels pour des domaines tels que le service à la clientèle (e.g. systèmes "help-desk" [RAC 97]), la gestion de connaissances (e.g. exploitation d'expériences [MIN 02]) et les applications du traitement de la langue (e.g. traduction automatique [MAC 00]).

4.1 FAQ-Finder – exploitation de questions-réponses

FAQFinder [BUR 95] est un système de questions-réponses basé sur les foires aux questions ("Frequently-Asked Questions" - FAQs) de USENET. Un FAQ est une réponse à une question fréquemment posée dans un groupe d'intérêt (e.g. groupe de programmation Java). Un FAQ est considéré comme un cas CBR car il contient la description d'un problème (la question) et la description d'une solution (la réponse). Un exemple de FAQ est présenté à la Figure 4.

[3] Une représentation de type *n-gramme* consiste à découper un texte en séquences de *n* caractères. Par exemple, le terme *raisonnement* serait représenté en trigramme par les séquences suivantes : {rai, ais, son, onn, nne, nem, eme, men, ent}.

FAQ # : 241
Question : *où se transigent les actions ordinaires de BCE ?*
Réponse : les actions ordinaires de BCE sont négociées aux Bourses de Toronto, New York ainsi qu'à la Bourse de Suisse sous le symbole BCE.

Figure 4. *Structuration de* **"***foires aux questions***"** *(***"***frequently-asked questions***"***)*

Le système est conçu pour recevoir en entrée une question en langage naturel et identifier les FAQs de USENET qui sont les plus similaires à cette question. La recherche de réponses pertinentes à ces questions est effectuée en 2 étapes.

La première étape permet de choisir, à partir de tous les fichiers FAQ de USENET, le sous-ensemble qui est le plus pertinent. Chaque fichier contient plusieurs dizaines de questions-réponses. Par exemple, à la question "What is garbage collection", on obtiendrait des fichiers de FAQ sur différents langages de programmation tels que Lisp et Java. Cette étape adopte une approche de recherche d'information et utilise le système SMART [BUC 85]. Les fichiers FAQ sont convertis selon le modèle vectoriel de recherche d'information et le rangement des fichiers est effectué selon des métriques statistiques ($tf*idf^4$). La comparaison entre la question et un fichier de FAQs est basée sur la correspondance exacte des termes des deux textes. Cette étape permet de filtrer les fichiers de FAQ et de n'en retenir que quelques dizaines.

La deuxième étape tente d'identifier, pour les fichiers jugés pertinents, les FAQs individuels qui correspondent le mieux à la question de l'usager. La correspondance entre la requête et chaque FAQ est évaluée selon trois métriques de similarité :

- métrique statistique : des fonctions du domaine de la recherche d'information sur des vecteurs de poids de type tf*idf. Différentes fonctions pour mesurer la distance entre les vecteurs de termes ont été testées dans leurs expérimentations.
- métrique sémantique : en utilisant le thésaurus WordNet, la distance sémantique entre chaque paire de mots est estimée. Pour évaluer cette distance, on utilise un algorithme de type "edge-counting" qui estime la distance entre deux concepts à partir du nombre de liens qui les séparent dans un réseau sémantique.
- métrique de recouvrement : pour quelques expérimentations, les auteurs ont tenté d'utiliser le pourcentage de mots de la requête qui est inclus dans les FAQs. Des résultats expérimentaux indiquent que cette métrique n'apporte pas de progrès significatifs et peut même causer une dégradation du système.

[4] Une mesure indiquant la fréquence d'un terme et sa capacité de discriminer entre plusieurs documents.

La similarité globale entre la requête et chaque FAQ est une somme pondérée de ces métriques. Les questions-réponses jugées les plus pertinentes sont présentées à l'usager en ordre décroissant de similarité. L'usager peut alors sélectionner les FAQs qu'il juge intéressants et recommencer la recherche.

Des expérimentations ont été menées sur un corpus de 241 questions, dont 138 avaient des réponses dans les FAQs de UseNet. Le système contient plus de 600 fichiers FAQ, et donc quelques dizaines de milliers de FAQs individuels. Les questions sont habituellement courtes (quelques dizaines de mots) mais les réponses sont plus longues et peuvent parfois contenir plus de mille mots.

La performance de la première phase, basée sur le système SMART, est excellente. Le bon fichier FAQ (i.e. le bon thème) est retourné parmi les cinq premières positions dans 88% des cas et en première position dans 48% des cas. La deuxième étape donne des résultats intéressants lorsque les métriques statistiques et sémantiques sont utilisées conjointement. La capacité du système à fournir une bonne réponse parmi les 5 premières recommandations est de 55% pour la métrique statistique seulement, 58% pour la métrique sémantique seulement, et 67% pour une métrique combinée. La qualité des résultats est toutefois limitée par l'utilisation de WordNet qui est une ressource linguistique trop générale pour ce type d'application. Egalement le système éprouve des difficultés à identifier les questions qui n'ont pas de réponses dans les FAQ.

Plusieurs tentatives ont été menées pour améliorer la performance du système [BUR 97]. L'analyse lexicale ("part-of-speech tagging") et syntaxique (grammaire hors-contexte) ont été utilisés pour identifier les termes importants des questions. Ces analyses n'ont pas permis d'améliorer significativement le système. Pour la deuxième étape du système, des techniques pour reconnaître le type de question et pour en faire la reconversion ont été appliquées. Les auteurs pensaient pouvoir améliorer les performances du système en restreignant les comparaisons entre questions de même type et en les exprimant sous différentes formes [BUR 97]. Par exemple, la question "Quand devrais-je changer l'huile de mon automobile?" peut être remplacé par la question "A quelle fréquence recommande-t-on de faire les changements d'huile?". Ceci correspond à paraphraser des questions à partir de canevas prédéterminés. Toutefois ces reconversions s'avèrent peu fiables lorsqu'elles sont effectuées uniquement à partir d'informations syntaxiques. Egalement un désambiguïseur sémantique a été utilisé pour améliorer la performance du système en terme de courbe rappel-rejet [LYT 00]. Finalement, des expérimentations ont été menées afin de déterminer automatiquement la pondération de chacune des fonctions de similarité par apprentissage automatique (programmation génétique) [COO 96] [BUR 97]. Par ailleurs, aucune tentative n'a été menée pour combiner ou modifier les réponses des FAQs (donc pas d'adaptation ou de modification de textes).

4.2 SPIRE - utilisation de cas pour rehausser la recherche d'information

Ces travaux, de J. Daniels et E. Rissland de l'Université du Massachusetts à Amherst [DAN 96] ont mené au développement de SPIRE, un système hybride de CBR et de recherche d'information. Dans ce système, le CBR aide les usagers du système de recherche d'information INQUERY à mieux formuler leurs requêtes et à identifier les passages pertinents dans les documents. Ainsi le module CBR agit comme pré-processeur et post-processeur pour une tâche de recherche d'information.

Le module CBR contient deux bases de cas : la première base contient des cas structurels décrivant les principaux attributs ("features") du contenu de documents tirés du corpus. La deuxième base contient des extraits textuels pour chacun des attributs d'un cas. Les bases de cas sont construites manuellement par un analyste humain.

Le traitement d'une requête s'effectue en deux étapes. Premièrement, la première base est utilisée pour sélectionner un nombre restreint de cas que l'on sait pertinents à la requête. Le contenu des cas est utilisé par le système INQUERY pour faire l'expansion de la requête. Ce mécanisme est analogue au "pseudo-relevance feedback" qui permet d'ajouter des termes à la requête initiale et d'ajuster le poids de chacun de ces termes. La requête étendue est alors traitée par INQUERY pour identifier les documents les plus pertinents de la collection.

La deuxième base de cas contenant les extraits sert à formuler une requête pour l'identification des passages. La requête contient soient tous les termes soient seulement les termes communs des passages reliés à un attribut. Cette requête est utilisée par INQUERY pour déterminer les fenêtres de mots pertinentes des documents retenus à la première étape. Une comparaison des extraits obtenus avec des requêtes formulées par le système et par un expert du domaine a été menée pour 10 attributs et 20 documents [DAN 98]. Les résultats indiquent que SPIRE offre une précision légèrement supérieure pour un plus grand nombre d'attributs (précision globale d'environ 50% pour chacun). Une analyse des résultats illustre l'avantage des extraits qui donnent un contexte plus diversifié que les requêtes formulées par des humains.

4.3 DRAMA – cas partiellement textuels

Le projet DRAMA [LEA 99][WIL 00] a pour but de gérer la connaissance des concepteurs de systèmes aéronautiques. Le système développé dans le cadre de ce projet aide les concepteurs lors du design de nouveaux avions et permet de préserver les différents aspects du design. Dans ce système, chaque design est

décrit à l'aide de cartes conceptuelles ("concept mapping"[5]), d'attributs descriptifs (e.g. caractéristiques du moteur) et d'annotations textuelles donnant des précisions sur les choix des concepteurs et sur les caractéristiques des composantes de l'avion. Puisque les cas contiennent des parties structurées (cartes et attributs valuées) et des parties non structurées (annotations), les auteurs les qualifient de "weakly-textual", i.e. des cas dont une partie est textuelle mais dont la plus importante portion est non-textuelle.

Le mécanisme de recherche de ce système combine des fonctions de similarité sur les attributs structurés et sur les annotations textuelles. Afin de simplifier la recherche textuelle, les auteurs utilisent un modèle vectoriel de recherche d'information. Plus précisément, la recherche comporte les étapes suivantes :

- chacun des attributs textuels est converti individuellement en vecteur de termes ; puisque les descriptions textuelles sont courtes, la sélection de termes ne repose pas sur la fréquence de mots-clé mais plutôt sur les syntagmes nominaux de type *Nom-Nom*, *Adj-Nom* ou *Nom-Prep-Nom*. Les syntagmes sont identifiés avec l'aide d'un lexique [WAR 94].
- un poids, similaire au tf*idf, est par la suite attribué aux syntagmes des différents vecteurs de termes.
- la similarité entre chaque paire de vecteurs textuels est déterminée selon la métrique du cosinus. Cette mesure est combinée aux similarités des autres attributs structurés des cas (cartes et attributs descriptifs).

En résumé, le système DRAMA illustre bien comment faire l'intégration de quelques attributs textuels au sein d'un système CBR structurel. Egalement il offre la particularité que la similarité entre les portions textuelles des cas est établie à partir des syntagmes nominaux. Bien que le système permette l'adaptation des diagrammes de design, les auteurs ne proposent pas de techniques pour adapter les portions textuelles de cas en fonction du nouveau design.

4.4 CBR-Answers – réseau pour la recherche de cas

Une approche pour améliorer la performance d'un système CBR est de structurer sa base de cas. Pour le système CBR-Answers, développé par Mario Lenz [LEN 97] [LEN 99], on utilise une structure de réseau pour "compiler" la base. Durant la phase de recherche, les valeurs de similarité sont propagées dans les nœuds du réseau et permettent de déterminer la pertinence de chacun des cas.

---

[5] Le "concept mapping" est un formalisme de représentation qui décrit par un graphe bi-dimensionnel la structure cognitive de la conception (les concepts et leurs interrelations). Contrairement aux réseaux sémantiques, les cartes conceptuelles ne sont pas contraintes syntaxiquement et n'ont pas de sémantique.

Tel qu'illustré à la figure 5, le réseau de recherche de cas ("case retrieval net") contient un ensemble de nœuds, les entités d'information (IE). Les IEs décrivent des éléments de documents (représentés par des rectangles arrondis) tels que des mots-clés, des termes complexes ("keyphrases"), des paires attributs-valeurs et des catégories du domaine. Ils décrivent également des identifiants de cas (représentés par des hexagones).

Les liens du réseau décrivent soit a) l'appartenance d'une entité d'information à un cas (liens continus), soit b) une relation de similarité entre deux entités (liens pointillés). Des poids qui indiquent le degré de similarité entre deux entités ou l'importance d'une entité par rapport à un cas sont attribués aux liens. Les liens représentent la structure d'inférence et guident la propagation des valeurs de similarité dans le réseau.

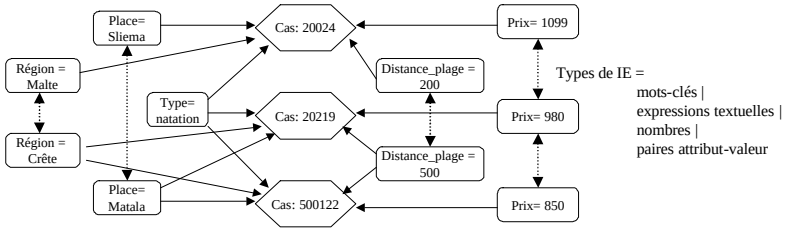

Figure 5. *Exemple de réseau de recherche de cas (adapté de [Lenz 98])*

Pour le traitement de cas décrit à partir de documents textuels, Lenz propose une procédure pour l'indexation des cas et la construction du réseau [LEN 98]. Cette structuration, qui repose sur une analyse lexicale des textes, est comme suit :

- chaque terme des documents est étiqueté selon sa catégorie lexicale et est normalisé en le remplaçant par sa racine morphologique. Les mots ayant une racine commune sont regroupés. Ces racines forment les traits des cas.
- les termes sont classés selon trois catégories (inutile, utile et potentiellement utile) et on retient ceux des deux dernières catégories. Les définitions proposées pour ces catégories sont :
  - inutile : des déterminants, des auxiliaires, des pronoms et des termes que l'on retrouve dans une liste de mots-outils.
  - utile : les adverbes, les adjectifs, les verbes et les noms.
  - potentiellement utile : les termes n'appartenant pas aux deux catégories précédentes.
- on complète la construction du réseau par une vérification manuelle des termes utiles et potentiellement utiles.

La recherche dans un réseau de recherche de cas débute par le découpage des termes de la requête en mots clé et leur association à des entités d'information du

réseau. Les entités sont ensuite activées. Les valeurs de similarité sont alors propagées parmi les différents IEs du réseau pour déterminer les IEs similaires. Finalement les valeurs de pertinence sont propagées aux nœuds des cas pour établir un ordonnancement parmi les différents cas de la base.

La mesure de similarité globale entre un nouveau problème Q et un cas C est estimée en fonction de leur similarité pour chacun des IE (*sim*) à l'aide de la fonction suivante :

$$SIM(Q,C) = \sum_{e_i \in Q} \sum_{e_j \in C} sim(e_i, e_j)$$

Le système CBR-Answers a été utilisé pour développer quelques applications commerciales de service à la clientèle dont FallQ (pour un fournisseur de services en télécommunication) et Simatic (pour le groupe Automation & Drive de Siemens AG). Ces systèmes permettent la recherche de documentation technique (e.g. spécifications de composantes, problèmes connus) et la réponse aux questions fréquentes (documents de type FAQ).

Des expérimentations menées avec FallQ, qui contient 45 000 documents, indiquent un temps de réponse qui varie entre 0.01 et 0.20 secondes par requête. Ce résultat illustre la bonne performance des algorithmes de propagation dans les réseaux de recherche de cas. Une étude a été menée avec Simatic pour évaluer la contribution des différents niveaux d'entités d'information [LEN 98]. Avec un corpus de 500 documents, l'utilisation de simples mots-clé offre la plus faible performance en terme de courbe précision-rappel et l'ajout successif de chaque couche d'entités (terme de thesaurus, de glossaire et thème du document) augmente significativement la performance du système. Cette étude illustre bien l'importance de la phase de structuration de documents lors de la création de la base de cas. Par contre, l'étude ne prend pas en compte l'ajout de paires attribut-valeurs, un élément important dans la structuration de cas semi-structurés.

4.5. *SMILE – Factorisation des cas par catégorisation de textes*

Ces travaux de Steffanie Brüninghaus [BRU 97] explorent des approches d'apprentissage automatique pour l'indexation des cas textuels. L'apprentissage permet de catégoriser des textes selon des attributs du domaine que les auteurs appellent "facteurs". Les textes sont des comptes-rendus d'actions en justice impliquant des fraudes de secrets industriels. Ces travaux ont été initiés par Aleven [ALE 96] qui a proposé un système tutoriel, basé sur le CBR structurel, pour enseigner aux étudiants de droit comment argumenter pour ce type de procès. Les "facteurs" représentent des situations qui jouent un rôle positif (favorable) ou négatif (défavorable) lors de l'argumentation de la cause. Les "facteurs" sont reliés selon une hiérarchie comprenant des niveaux spécifiques, des niveaux abstraits et des niveaux de thèmes généraux.

Dans les travaux d'Aleven, les cas étaient indexés manuellement. Les travaux sur SMILE visent à extraire automatiquement des facteurs à partir de ces textes légaux. On note trois séries de travaux :

- *la catégorisation de textes complets* [BRU 97] : chaque texte est représenté comme un vecteur de fréquence de mots-clé. Des expérimentations ont été menées pour catégoriser un corpus de 147 cas selon 26 facteurs à l'aide de techniques d'apprentissage tel que "Naïve Bayes", "Winnow" , "Rocchio" et "Exponentiated Gradient". Pour la plupart des facteurs, les algorithmes ont identifié peu d'exemples positifs, amenant des faibles performances en terme de justesse ("accuracy"), de précision et de rappel.

- *la catégorisation de passages* [BRU 99] : divers passages sont étiquetés manuellement pour indiquer la présence de facteurs dans le texte (voir figure 6). Ces passages servent de corpus d'entraînement pour apprendre comment catégoriser les portions de textes. Un thésaurus est également utilisé pour identifier les correspondances entre des termes de différents passages. Ceci permet de détecter la présence de facteurs exprimés avec des mots différents mais significativement équivalents. A partir d'un échantillon de 2200 passages (de longueur moyenne de 7.5 termes), l'algorithme ID3 a atteint jusqu'à 80% de précision et de rappel (minimum : 30% de précision et 50% de rappel). L'utilisation du thesaurus est par contre moins concluante. Pour certains facteurs, elle apporte des améliorations de 40% tandis qu'elle apporte une dégradation de 10% à 20% pour d'autres facteurs.

- *la catégorisation à partir d'informations extraites* [BRU 01] : plus récemment, Brüninghaus a proposé d'utiliser le système d'extraction d'information AutoSlog [RIL 96] pour repérer trois types d'information : les entités nommées, les "case frames", et les négations. Les informations extraites seraient alors utilisées par le processus d'apprentissage pour identifier la présence de facteurs dans les textes légaux. Ces travaux sont en cours et aucune expérimentation n'a encore été menée.

Case 37
<F15/> Plaintiff's packaging process involved various "tempering steps" that were not used by competitors or described in the literature. </F15> <F16/> Only a handlful of plaintiff's employees knew of the packaging operations, and they were all bound by secrecy agreements. </F!6>. <F6/> There was also testimony that packaging information was closely guarded in the trade.</F6>
<F1/>Plaintiff's president sent a lette to defendant which conveyed plaintiff's manufacturing formula. </F1>. <F21/> The letter also stated….

Figure 6. *Étiquetage de passages selon des facteurs (tirée de [Bru 99])*

4.6 PRUDENCIA – structuration de cas de type «template mining»

PRUDENCIA est un système qui facilite la recherche documentaire en jurisprudence légale [WEB 98a][WEB 98b]. Il permet de rechercher des situations, décrites dans des documents textuels, qui sont similaires à une nouvelle cause juridique.

La principale contribution de ces travaux est de proposer une démarche pour convertir des textes légaux en cas structurés. Cette démarche s'appuie principalement sur la forte structuration des documents légaux utilisés dans ce projet. On retrouve dans chaque texte un certain nombre de sous-sections (e.g. "entête", "résumé", "corps", "conclusion"). Les sous-sections comportent une régularité puisque leur contenu est homogène (mêmes thèmes) et qu'on y retrouve des phrases identiques situées aux même endroits. Cette régularité facilite le processus d'extraction du contenu des textes.

Avant d'être exploités par un système CBR, les documents sont structurés à l'aide de formulaires comprenant neuf champs (type de pétition, numéro de cas, district, page, date, fondation, thème, lois secondaires, catégorie, résultat, unanimité). Les champs des formulaires et leurs valeurs admissibles ont été sélectionnés par un expert du domaine. Le processus de structuration des cas reposent sur un certain nombre de méthodes qui alimentent les formulaires. On note les méthodes suivantes :

- directe : des attributs sont explicites et leurs valeurs sont situées à des positions fixes dans le texte ;
- par mots-clé : pour chaque champ du formulaire, on recherche dans les sous-sections du texte des mots-clé contenus dans une liste d'expressions. Pour faciliter la recherche, la racine des mots et un dictionnaire de synonymes sont utilisés ;
- par patron : des expressions régulières permettent d'obtenir des numéros d'articles de loi (e.g. "for infringing articles 26 & 97 of Penal Code"). Les patrons sont définis manuellement ;
- par règle : des règles permettent de tenir compte de la dépendance entre différents champs du formulaire et de diriger ainsi la méthode vers la sous-section adéquate. Les règles sont définies manuellement ;
- par comparaison : le champ "thème" est déterminé suite à une comparaison avec d'autres cas structurés (similaire à une classification par les plus proches voisins).

Le système contient 3500 cas de jurisprudence. La recherche se fait par comparaison entre les champs des formulaires à l'aide de métriques binaires (0 ou 1) et graduées (sur une échelle [0,1]). Quelques expérimentations permettent d'évaluer le temps de recherche du système, mais aucune évaluation du processus de recherche en terme de précision et rappel n'est présentée.

5. Comparaison des travaux en CBR textuel

Les deux tableaux suivants résument les principaux points des travaux présentés à la section précédente. Le Tableau 1 décrit les particularités de la tâche accomplie par le système et le Tableau 2 présente les particularités techniques de l'approche CBR préconisée.

On note qu'une majorité de systèmes sont utilisés pour des tâches de type "recherche d'information" (IR) : DRAMA, FAQFinder, CBR-Answers, SPIRE et PRUDENCIA. Par contre, la plupart de ces applications se démarquent des approches typiques de recherche d'information par l'utilisation de connaissances du domaine dans la structuration de la base de cas et dans la prise en compte de la tâche à accomplir dans le processus de recherche (mesures de similarité du domaine). Une autre différence par rapport aux applications de recherche d'information est que ces applications reposent sur la recherche d'un seul cas similaire pour accomplir leur tâche. Il n'est donc pas souhaitable que ces applications retournent le plus de cas pertinents possibles. En fait, plusieurs bases de cas de ces systèmes ne contiennent que des cas pertinents dont le degré de similarité varie.

L'étendue des domaines d'application de ces systèmes varie. Les approches de SMILE, SPIRE et PRUDENCIA présument que des connaissances du domaine sont disponibles pour la sélection des attributs et la structuration des cas. En revanche, il y a peu de contraintes pour FAQFinder car on ne tente pas de modéliser les domaines abordés par les FAQs de USENET.

On note que le degré de structuration et la longueur des textes varient également. La plupart des textes utilisés dans ces applications sont peu structurés. Certains textes n'offrent pas de découpage (e.g. SMILE, CBR-Answers), d'autres se limitant à distinguer les portions « problème » et « solution » des cas (les FAQs de FAQFinder). Seuls les textes juridiques de PRUDENCIA offrent un grand nombre d'attributs explicitement décrits dans les textes. La petite taille du corpus et la complexité des textes représentent un défi pour SMILE, ce qui explique l'approche distincte qui y est utilisée.

Le Tableau 2 illustre qu'aucun des projets présentés ne propose de processus complexes dans le choix des traits de cas. Soit que le choix est fait manuellement (SMILE, SPIRE, PRUDENCIA), soit statistiquement par des méthodes du domaine de la recherche d'information (DRAMA et FAQFinder). La démarche proposée par CBR-Answers est mixte. Pour les approches statistiques, les textes font l'objet d'étiquetage lexical et d'extraction de racine.

Travaux	Tâche	Caractéristiques du domaine	Caractéristiques des cas
FAQFinder	*Recherche* de documents structurés selon un format question-réponse (FAQ).	Pas de domaine en particulier. Tout texte de type "frequently-asked questions" peut être considéré, indépendamment du sujet du groupe d'intérêt.	Les fichiers initiaux (~600) sont longs, contenant plusieurs centaines de FAQs. Un FAQ est structuré en deux parties : une courte question, et une réponse comptant quelques centaines de mots.
Drama	*Recherche* de dossiers de design et préservation de la connaissance des concepteurs.	Aéronautique, ce qui laisse présager un vocabulaire restreint et un groupe de concepts relativement limités.	Les 62 cas sont en partie structurés (diagrammes, attribut-valeur) et en partie textuels. Les attributs textuels sont courts et moins importants que les attributs structurés.
Spire	*Recherche* de passages pertinents dans un corpus de documents.	Gestion de faillite personnelle. L'approche ne dépend pas du domaine. Applicable à un domaine restreint seulement.	Aucune contrainte sur la nature des textes. Une base de cas décrit des documents du corpus (cas structurels) et l'autre base contient des extraits textuels (moyenne : 46 termes significatifs).
CBR-Answers	Help-desk pour la *recherche* de documents question-réponse et de documents corporatifs (gestion de connaissance).	Domaines d'automatisation de processus et télécommunications. Exploitation du vocabulaire et des concepts du domaine. Approche valide pour les domaines peu restreints.	Seuls les documents question-réponse sont structurés. La longueur des documents varie. La base de l'application FallQ contient 45 000 cas.
Smile	*Catégorisation* de textes légaux selon différents facteurs.	Domaine de la fraude relié aux secrets industriels. Basés sur une terminologie et des concepts propres aux domaines.	Des textes légaux relativement longs et complexes. Aucune structuration des textes à priori. La base contient 147 cas.
Prudencia	*Recherche* de textes légaux.	Domaine de la jurisprudence légale.	Des textes légaux relativement complexes (moyenne de 725 mots). Prise en compte la structure rhétorique des textes. La base contient 3500 cas.

Tableau 1. Caractéristiques des domaines des systèmes CBR textuel

Travaux	Indexation	Structuration	Recherche
FAQFinder	Mots-clés, provenant des fichiers, ayant fait l'objet de stemming et filtrés par rapport à une liste de mots-outils.	Création automatique de vecteurs de mots-clés et attribution de poids (tf*idf).	Style IR avec cosinus. Utilise des métriques de similarité statistique(tf*idf) et de similarité sémantique (edge-counting). Reformulation de questions (paraphrasage) pour faciliter la comparaison.
Drama	Sélection de syntagmes nominaux tirés des textes à l'aide d'étiquetage lexical.	Création automatique de vecteurs de termes (syntagmes) et attribution de poids (tf*idf).	Style IR avec opérateur de cosinus.
Spire	Un groupe d'attributs ("features") du domaine sélectionnés par le concepteur.	Création manuelle de "frames" et d'extraits textuels.	Comparaison d'attributs pour le CBR et recherche IR par le système INQUERY.
CBR-Answers	Mots-clés de fichiers, syntagmes nominaux et termes du domaine ajoutés manuellement par le concepteur.	Création d'un réseau qui relie i) les attributs entre eux et ii) les cas aux attributs.	Propagation des valeurs de similarité dans un graphe dirigé («case retrieval net»).
Smile	Un groupe de facteurs provenant d'une modélisation manuelle du domaine.	Catégorisation des documents, de passages ou d'informations extraites à partir de techniques d'apprentissage automatique.	Comparaison de la présence/absence de facteurs (décrit dans [ALE 97]).
Prudencia	Des attributs fournis par un expert du domaine juridique.	Création semi-automatique de "frames" (template mining)	Comparaison d'attributs avec mesures binaires et graduées.

Tableau 2. *Caractéristiques techniques des systèmes CBR textuel*

Il est intéressant de noter le niveau de structuration des cas par rapport au mécanisme de recherche de chacun des systèmes. Les systèmes FAQFinder et SPIRE misent principalement sur des mécanismes plus élaborés de recherche pour accomplir leur tâche. Les systèmes SMILE et PRUDENCIA axent plutôt leurs efforts sur un enrichissement des cas pour atteindre de bonnes performances. CBR-Answers est le seul système à oeuvrer sur les deux plans.

Quelle est la meilleure approche ? Devrait-on miser sur une représentation conceptuelle de cas, sur des fonctions de similarité plus riches ou sur un formalisme de recherche plus élaboré ? Bien que l'on puisse identifer les avantages et désavantages de chaque système (voir Tableau 3), il ne se dégage malheureusement

pas de réponse à cette question à partir des travaux répertoriés. Toutefois plusieurs facteurs jouent un rôle important dans le choix de l'approche.

Travaux	Pour	Contre
FAQFinder	Une combinaison fructueuse de similarités statistique et sémantique.	Le système éprouve des difficultés à identifier les questions sans réponse
Drama	La simplicité et l'intégration des attributs structurels et textuels.	Les syntagmes nominaux apportent-ils une contribution significative?
Spire	Les extraits de texte permettent de bien définir le contexte de recherche.	L'approche se transpose mal dans un cadre de résolution de problème.
CBR-Answers	La rapidité de la phase de recherche.	Il y a des limitations sur la nature des métriques de similarité.
Smile	L'approche de structuration de cas par apprentissage est efficace.	L'utilisation de connaissances sémantiques semble inefficace.
Prudencia	On y exploite la forte structuration des textes du domaine applicatif.	L'approche est limitée aux domaines restreints et la méthodologie de structuration manque de généricité.

Tableau 3. *Avantages et désavantages des systèmes CBR textuel*

Il semble que la complexité/longueur des textes et l'étendue du domaine soient les principaux facteurs à considérer. La diversité du domaine rend difficile l'acquisition de connaissances du domaine et favorise donc l'utilisation de techniques de recherche plus élaborées. Une structuration naturelle des textes facilite l'utilisation de métriques de similarité plus complexes. La concentration de la base de cas dans un domaine pointu avec un vocabulaire restreint oblige le concepteur à indexer et à structurer avec précision chacun des cas.

Les travaux que nous avons présentés ne nous permettent pas de bien cerner l'influence de ces facteurs. Les applications construites à partir des textes longs portent sur des domaines restreints (SMILE, SPIRE, PRUDENCIA), ce qui favorise une structuration plus élaborée des cas. Et les textes des applications à domaine plus vaste (CBR-Answers et FAQFinder) sont des FAQs plutôt courts. Or, de plus amples recherches devront être effectuées pour tenter de quantifier chacune de ces dimensions pour des corpus de textes mieux diversifiés.

Bien que déjà mentionné à quelques reprises, il est important de souligner qu'aucun système CBR textuel ne fait d'adaptation de texte. Il y a lieu de se demander pourquoi. Le peu de structuration des solutions dans les documents est habituellement invoqué comme la principale raison motivant l'absence d'adaptation. On pourrait également affirmer que la nature des tâches à accomplir est une autre limitation. Pour plusieurs applications de type "recherche d'information", il est préférable de repérer le cas qui satisfait la tâche à accomplir et de laisser l'usager extraire les informations qui répondent à ses besoins. D'autres applications ne se prêtent pas naturellement à l'adaptation ; par exemple,

les applications de jurisprudence. Celles-ci visent à identifier les avis légaux qui peuvent être utilisés pour appuyer un plaidoyer et non pas à rédiger une nouvelle décision légale. Finalement des considérations techniques sont à considérer. L'adaptation de textes devrait reposer sur des techniques de linguistique informatique pour l'analyse syntaxique/sémantique et la génération des textes. Or il y a lieu de croire que la communauté CBR ne maîtrise pas actuellement les outils nécessaires pour aborder ces tâches.

6. Travaux futurs

Le CBR textuel est relativement récent et plusieurs avenues de recherche méritent d'être explorées afin de mieux définir ses frontières et de quantifier ses approches. Contrairement aux modèles structurel et conversationnel, il n'existe pas de vision unifiée du modèle CBR textuel mais plutôt un ensemble de travaux disparates. Une telle vision unifiée permettrait de mieux regrouper les différents travaux et de déterminer les principales lacunes du modèle. Le domaine présente également certaines déficiences d'un point de vue méthodologique. Tel que mentionné dans la section précédente, des expérimentations sont nécessaires pour obtenir une meilleure caractérisation des facteurs influençant le choix d'une approche particulière.

Nous présentons dans les prochains paragraphes un certain nombre de points qui, à notre connaissance, n'ont pas fait l'objet de recherche en CBR textuel.

Construction de cas ("authoring") : c'est probablement à ce niveau que les contributions les plus significatives peuvent être apportées. La littérature CBR actuelle propose des représentations de cas comportant des mots-clé, des termes complexes, des catégories et des paires attributs-valeurs. Toutefois, pour une application particulière, il n'existe pas de méthodologie pour déterminer le niveau de représentation adéquat ni de critères pour faire ce choix.

Cette lacune risque de s'avérer encore plus importante avec l'avènement du web sémantique. Nous faisons référence plus particulièrement à la construction de bases de cas à partir de documents semi-structurés ayant fait l'objet d'annotations. Des approches seront nécessaires pour déterminer l'importance relative du contenu textuel par rapport à sa contextualisation sémantique et pour les mettre à contribution dans le processus de résolution de problème. Pour les documents non-structurés, les techniques d'extraction d'information et de fouille de textes ("text mining") pourraient jouer un rôle important dans le processus de structuration de cas.

Recherche sur la base de cas : bien que ce thème ait largement été étudié, certaines idées restent à explorer. Premièrement, plusieurs des méthodes actuelles préconisent une représentation de type "mots en vrac" ("bag of words") ne tenant pas compte de l'ordre des termes d'indexation. Or, pour des applications de

résolution de problèmes, la préservation d'informations linguistiques comme la négation de propositions ou des séquences particulières de mots peut jouer un rôle sur le choix des solutions proposées.

Une approche prometteuse, surtout pour des domaines restreints ayant des corpus de grande taille, serait l'utilisation de modèles statistiques de langue pour estimer la similarité entre des textes. Également, la compression de cas telle que préconisée par des méthodes comme le "Latent Semantic Indexing" [DEE 90] [HOF 99] pourrait permettre d'obtenir de meilleures représentations internes de cas. Finalement, l'enrichissement de ressources linguistiques (telles que WordNet) par des techniques de regroupement ("clustering") de mots pourrait favoriser la définition de métriques sémantiques pour de nouveaux domaines applicatifs.

Adaptation : l'adaptation de solutions textuelles est une tâche ardue pour laquelle ni paradigme, ni modèle, ni approche n'ont été proposés jusqu'à maintenant. La communauté CBR y prête peu attention en raison du manque d'incitatifs économiques et des difficultés techniques qu'elle comporte. Mais le problème demeure important d'un point de vue académique et pour des applications comme le service personnalisé à la clientèle.

Une approche possible, de type transformationnelle, est de mener l'adaptation comme une séquence de "réparations" ("iterative repair") appliquées à une solution textuelle. Ce processus peut être mis en œuvre à l'aide de deux opérations : identifier les portions de texte devant être modifiées et déterminer comment les modifier.

Pour le premier type d'opération, la pertinence des différentes portions d'une solution textuelle doit être établie en fonction d'une nouvelle description de problème. Des techniques permettant d'évaluer le recouvrement thématique des textes et la spécificité des informations (e.g. description de lieux, individus, dates…) pourraient être utilisées. Le recouvrement thématique est la correspondance entre les portions d'un texte "solution" et un texte "problème". Tout comme la similarité, cette notion fait appel à une mesure de pertinence. Il serait donc intéressant d'établir le lien entre les connaissances nécessaires à cette étape d'adaptation et celles utilisées pour la phase de recherche. Il y a même lieu de croire que ces connaissances pourraient être les mêmes.

Le deuxième type d'opération repose plutôt sur une analyse du domaine. Les travaux sur la définition de règles d'adaptation en CBR textuel et sur l'apprentissage de règles d'association pourraient servir pour les domaines offrant un fort niveau de prévisibilité.

Maintenance : la maintenance de bases de cas est habituellement guidée par des critères tels que la redondance entre cas ou le recouvrement des solutions ("coverage" et "reachability"). Ces critères se prêtent bien au CBR structurel car les cas y sont structurés sans ambiguité, les attributs et leur domaine étant définis à partir d'une description du domaine. Toutefois, pour appliquer ces approches de

maintenance à des cas textuels, il serait important d'étendre ces critères pour tenir compte de la synonymie et la polysémie des différents termes indexant les cas.

Nous explorons actuellement certaines de ces avenues de recherche dans le cadre d'un projet de réponse automatique au courrier électronique [LAM 01]. Cette tâche se prête bien au CBR textuel car les entreprises disposent de corpus (messages antécédents, "FAQ", documents et glossaires) permettant de constituer une base de cas représentative du domaine applicatif. Toutefois la tâche de réponse automatique au courriel présente de nombreux défis. Pour être utile, la phase de recherche du système CBR doit offrir une forte précision. Il est donc nécessaire de mettre l'accent sur la démarche d'authoring de la base de cas pour obtenir des performances nettement supérieures aux résultats présentés dans la littérature. De plus, les messages comportent souvent plus d'un thème et plus d'une requête. Cette particularité exige la capacité soit de fragmenter les cas ou de mener des recherches multiples. Finalement la tâche de réponse automatique exige une modification des réponses antécédentes en fonction de la nouvelle requête. Ces adaptations comprennent la personnalisation des messages (substitution d'entités nommées comme le nom d'individu, l'adresse, le nom de l'entreprise…) et l'élagage des portions de messages qui ne sont plus pertinentes au nouveau contexte (pour les messages à thèmes et à questions multiples).

7. Conclusion

L'exploitation d'expériences décrites dans des documents textuels a suscité depuis quelques années l'intérêt de nombreux chercheurs de l'apprentissage automatique, de la recherche d'information et de la gestion de connaissance. La communauté CBR, qui en est à ses premiers balbutiements dans cette voie, propose des approches qui tentent de concilier la structuration de textes et des schémas de recherche qui tirent profit de caractéristiques du domaine applicatif. Nous avons, dans cet article, présenté les principales contributions de cette communauté.

Bien qu'aucune des approches actuelles du CBR textuel ne prédomine, nous croyons que cette voie est prometteuse car son objectif à long terme de résoudre des problèmes à partir de descriptions textuelles lui permet de se démarquer des autres domaines connexes. Cette facette de résolution de problème a toutefois été négligée jusqu'à maintenant, principalement par l'absence de travaux sur l'adaptation textuelle et par la moindre importance accordée au traitement des solutions textuelles dans les phases du cycle de raisonnement. Comme directions futures, une étude comparative des différentes approches ainsi qu'une meilleure exploitation des techniques de traitement de la langue naturelle permettront de faire avancer l'état actuel du domaine et de bien établir ses fondements de recherche.

Remerciements

La réalisation de cette recherche a été rendue possible grâce à la participation financière de Bell Canada à travers son programme de support à la R-D des Laboratoires universitaires Bell.

8. Bibliographie

[AAM 94] Aamodt A., Plaza E., « Case-base reasoning : foundational issues, methodological variations, and system approaches », AI Communications, vol. 7, no. 1, 1994, p. 39-59.

[AHA 01] Aha D.W., Breslow L.A., Muñoz-Avila H, « Conversational case-based reasoning », Applied Intelligence, vol 14, 2001, p. 9-32.

[ALE 96] Aleven V., Ashley K. D., « How Different is Different? Arguing about the Significance of Similarities and Differences », Proceedings of *EWCBR-96*, LNAI 1168, Springer Verlag, 1996, p. 1-15.

[BRA 96] Branting L. K., Lester J., « Justification Structures for Document Reuse », Proceedings of *EWCBR-96*, LNAI 1168, Springer Verlag, 1996, p. 76-90.

[BRU 01] Brüninghaus S., Ashley K. D, « The Role of Information Extraction for Textual CBR », Proceedings of *ICCBR-01*, LNAI 2080, Springer Verlag, Berlin, p. 74-89.

[BRU 99] Brüninghaus S., and Ashley K. D., « Bootstrapping Case Base Development with Annotated Case Summaries », Proceedings of *ICCBR-99*, LNCS 1650, Springer Verlag, 1999, p. 59-73.

[BRU 97] Brüninghaus S., Ashley K. D., « Finding Factors: Learning to Classify Case Opinions Under Abstract Fact Categories », Proceedings of *ICAIL-97*, 1997, p. 123-131.

[BUC 85] Buckley C., « Implementation of the SMART Information Retrieval System », Rapport technique 85-685, Université Cornell, 1985.

[BUR 97] Burke R., Hammond K., Kulyukin V., Lytinen S., Tomuro N., Schoenberg S., Question Answering from Frequently-Asked Question Files: Experiences with the FAQ Finder System, Rapport technique TR-97-05, Université de Chicago, Départment d'informatique, 1997.

[BUR 95] Burke R., Hammond K., Kozlovsky J., « Knowledge-based Information Retrieval for Semi-Structured Text », Working Notes from AAAI Fall Symposium on *AI Applications in Knowledge Navigation and Retrieval*, AAAI, 1995, p. 19-24.

[COO 96] Cooper E., « Improving FAQfinder's Performance: Setting Parameters by Genetic Programming », AAAI Spring Symposium on Machine Learning in Information Access, Stanford, 1996.

[DAN 98] Daniels, J., Risland, E., « Locating Passages Using a Case-Base of Excerpts », *Seventh International Conference on Information and Knowledge Management (CIKM '98)*, Washington, 1998.

[DAN 96] Daniels J., Retrieval of Passages for Information Reduction, Thèse de doctorat, Université du Massachusetts, Amherst, 1996.

[DEE 90] Deerwester S., Dumais S. T., Furnas G. W., Landauer T., Harshman R., « Indexing by Latent Semantic Analysis », *Journal of the American Society for Information Science*, vol. 41, no. 6, p. 391–407, 1990.

[FUC 00] Fuchs B., Lieber J., Mille A., Napoli A., « An algorithm for adaptation in case-based reasoning. », Proceedings of *ECAI'2000*, Amsterdam, p. 45-49, 2000.

[FUC 99] Fuchs B., Lieber J., Mille A., Napoli A., « Vers une théorie unifiée de l'adaptation en raisonnement à partir de cas. », Actes de *RàPC-99*, Palaiseau, France, p. 77-85, 1999.

[HOF 99] Hofmann T., « Probabilistic Latent Semantic Indexing. », Proceedings of the *22nd International Conference on Research and Development in Information Retrieval (SIGIR'99)*, 1999.

[KOL 93] Kolodner J., *Case-Based Reasoning*, Morgan Kaufmann, 1993.

[LAM 01] Lamontagne L., Raisonnement à base de cas textuel pour la réponse automatique au courrier électronique, Rapport interne, Université de Montréal, 2001.

[LEN 99] Lenz M., Glintschert A., « On Texts, Cases, and Concepts. », Proceedings of *XPS-99*, LNAI 1570, Springer Verlag, p. 148-156, 1999.

[LEN 98] Lenz M., Bartsch-Spörl B., Burkhard H.-D., Wess S. (Eds.), *Case-Based Reasoning Technology - From Foundations to Applications*, LNAI 1400, Springer Verlag, 1998.

[LEN 97] Lenz M., Burkhard H.-D., « CBR for Document Retrieval - The FAllQ Project », Proceedings of *ICCBR-97*, LNAI 1266, Springer Verlag, p. 84-93, 1997.

[LEA 99] Leake D. B., Wilson, D. C., « Combining CBR with Interactive Knowledge Acquisition, Manipulation and Reuse », *Proceedings of ICCBR-99*, LNAI 1650, Springer-Verlag, p. 203-217, 1999.

[LEA 96] Leake D. B. (éditeur), *Case-Based Reasoning: Experiences, Lessons, and Future Directions*, AAAI Press/MIT Press, Menlo Park, CA, 1996.

[LEA 01] Leake D. B., Smyth B., Yang Q., Wilson D., Special Issue on Maintaining Case-Based Reasoning Systems, Computational Intelligence, Vol. 17, No. 2, 2001.

[LYT 00] Lytinen S., Tomuro N., Repede, T., « The Use of WordNet Sense Tagging in FAQFinder », Proceedings of the *Workshop on Artificial Intelligence for Web Search*, AAAI-2000, Austin, TX, 2000.

[MAC 00] Macklovitch E., Simard M., Langlais P., « TransSearch: A Free Translation Memory on the World Wide Web », Proceedings of *Second International Conference On Language Resources and Evaluation*, Athens, Greece, p. 1201-1208, 2000.

[MAN 99] Manning C., Schütze H., *Foundations of Statistical Natural Language Processing*, The MIT Press, Cambridge, Massachusetts, 1999.

[MIN 02] Minor M., Staab S. (éditeurs), *1st German Workshop on Experience Management*, Lecture Notes in Informatics P-10, Bonner Köllen Verlag, 2002.

[RAC 97] Racine K., Yang Q., « Maintaining unstructured case bases », *Proceedings of the Second International Conference on Case-Based Reasoning ICCBR-97*, LNAI 1266, p. 553-564, 1997.

[RIE 89] Riesbeck C., Shank R., *Inside Case-Based Reasoning*, Lawrence Erlbaum Associates, 1989.

[SAL 83] Salton G., McGill M. J., *Introduction to Modern Information Retrieval*, McGraw-Hill, 1983.

[SAL 88] Salton G., Buckley C., « Term-Weighting Approches in Automatic Text Retrieval », *Information Processing and Management*, vol. 24, p. 513-523, 1988.

[SMY 95] Smyth B., Keane M. T., « Experiments on Adaptation-Guided Retrieval in a Case-Based Design System », Proceedings of *ICCBR-95*, LNAI 1010, Springer-Verlag, p. 313-324, 1995.

[WAR 94] Ward G., « Grady Ward's Moby Lexicon », http://www.dcs.shef.ac.uk/research/ilash/Moby/ , University of Sheffield, 1994.

[WAT 98] Watson I., *Applying Case-Based Reasoning: Techniques for Enterprise Systems*, Morgan Kaufmann Publishers Inc., 1997.

[WEB 98a] Weber R., Martins A., Barcia R, « On legal texts and cases », *Textual Case-Based Reasoning: Papers from the AAAI-98 Workshop*, Rapport technique WS-98-12, AAAI Press, p.40-50, 1998.

[WEB 98b] Weber, R., Intelligent Jurisprudence Research, Thèse de doctorat, Université fédérale de Santa Catarina, Brésil, 1998.

[WIL 00] Wilson D. C., Bradshaw S., « CBR Textuality », Expert Update, Vol. 3., No. 1, pp. 28-37, 2000.