UNIVERSITÉ DE GENÈVE

Favoriser la perception des communautés en ligne et la diffusion des connaissances en résumant les informations publiées sur des pages personnelles

Mélie Genêt

Mémoire présenté pour l'obtention du DESS STAF
"Sciences et Technologies de l'Apprentissage et de la Formation"

TECFA,
Faculté de Psychologie et de Sciences de l'Education
Université de Genève

Juillet 2005

Jury :

Daniel Schneider	MER, TECFA	Directeur
Mireille Bétrancourt	Professeure, TECFA	Examinateur
Nicolas Nova	Collaborateur scientifique EPFL	Examinateur

Table des Matières

Abstract

Friend-Of-A-Friend (FOAF) est un langage dérivé du XML qui permet de décrire les personnes et leurs liens, et de manipuler ces données avec une grande facilité. Il s'impose donc comme une référence pour l'étude des réseaux sociaux. Cependant, la plupart des gens négligent de créer des métadonnées les concernant et préfèrent publier des informations non structurées sur leurs pages personnelles. Nous faisons la proposition d'un système de génération automatique de profils FOAF à partir des données des pages personnelles, qui va produire une banque de données interrogeable sous forme visuelle et textuelle. Le programme utilise par ailleurs une technique statistique pour détecter des termes sémantiquement proches et déceler des communautés d'intérêts au sein des membres du TECFA *(Technologie de Formation et d'Apprentissage, unité de la FPSE, Université de Genève)* : l'analyse de sémantique latente. Les prototypes fournis, quoi que nécessitant des développements ultérieurs importants, présentent de nombreux aspects novateurs et pourraient être utilisés afin de promouvoir la perception de communautés de pratiques, de supporter la localisation d'experts et de soutenir le travail collaboratif.

Remerciements

Nous tenons à remercier le professeur Daniel Schneider sans qui ce travail n'eût pas été possible. Il nous a aidés à de ne nombreuses reprises pour résoudre certains problèmes méthodologiques et techniques très certainement insurmontables sans son appui. Nous le remercions notamment pour son adaptation de l'algorithme LSA de Bellcore qui n'aurait pu être installé sur le serveur du TECFA avec les versions actuelles de GCC. Nous remercions Messieurs Benoit Lemaire (Laboratoire LEIBNIZ-IMAG, Grenoble) et Philippe Dessus (UPMF, Grenoble) pour leur aide quant à l'analyse de sémantique latente. Nous remercions l'équipe du TECFA pour leur soutien au cours de nos recherches. Nous remercions Monsieur Sébastien Monnaert pour sa tolérance et son aide. Enfin nous rendons hommage à Monsieur Jean-Jacques Duclaux qui nous a épaulés au long de notre travail.

1. Introduction

Friend-Of-A-Friend (FOAF) est un langage machine qui permet de décrire les personnes et leurs réseaux de relations. Il fournit un vocabulaire pour décrire qui l'on est, ce qui nous intéresse et quels sont nos amis. Certains l'ont utilisé pour mettre en ligne leurs profils personnels, pointant à leurs tours vers les profils de leurs connaissances. Ce réseau de fichiers FOAF publiés sur le Web constitue un véritable « réseau social », directement compréhensible par les machines. Cependant, ces informations restent peu nombreuses, sans commune mesure avec les millions de données recueillies par certaines applications sociales (« *social softwares* ») propriétaires, telles *Friendster* ou *Orkut*.

Le langage FOAF, et plus généralement les initiatives du W3C (*World Wide Web Consortium*) pour promouvoir un Web sémantique, sont boudés par le grand public parce qu'ils exigent un investissement de temps pour créer des métadonnées. Le manque d'une application de vulgarisation auprès du grand public se fait encore sentir. MIKA (2002), a élaboré un système qui collectait les profils FOAF et utilisait des techniques de « Web-mining » pour extraire des informations sur les réseaux sociaux à l'aide des données publiées sur le Web. Il a ensuite démontré comment ces informations pouvaient donner des indices sur une communauté, en empruntant des méthodes à l'analyse de réseaux sociaux (« social network analysis »), une branche de la sociologie concernée par les données relationnelles. A son instar, nous avons envisagé Internet comme une source de connaissances sur les gens et les communautés. Le langage FOAF nous est apparu comme une solution particulièrement aisée pour stocker et manipuler les données sur les gens. Dans cette optique, nous avons cherché à développer une interface qui permette de générer automatiquement des profils sur les étudiants de l'université à partir des données, structurées ou non, publiées sur leurs pages personnelles. Nous souhaitions que l'interface permette d'interroger les profils générés et de les afficher sous forme visuelle. Nous n'avions pas pour ambition d'égaler la qualité de données fournies par les personnes ayant créé leurs propres profils FOAF, mais nous espérions générer des fichiers suffisamment proches de la réalité pour permettre de donner des indices sur les communautés de pratiques et d'intérêts, et les préoccupations des membres du TECFA, le département de l'université de Genève auquel nous appartenons. Comment mettre en lumière des réseaux sociaux sur le Web à l'aide de la matière contenue dans les pages personnelles ? Comment localiser des personnes qui partagent un intérêt ? De nombreux systèmes de gestion des connaissances proposent déjà des systèmes de localisation des puits de compétences, mais ils nécessitent un recueil d'informations long et onéreux. Une fois la personne compétente ciblée, ces programmes ne donnent pas d'indications précises sur elle, sur son degré de proximité ni sur la manière de la contacter.

Dans le cadre de ce travail de diplôme, nous faisons la proposition d'un système de génération automatique de profils FOAF, qui va constituer une banque de données interrogeables sous forme textuelle ou visuelle, permettant la mise en évidence de caractéristiques communes entre les membres du TECFA (Université de Genève), et la localisation de communautés d'intérêt et de pratiques. Nous commençons par situer notre travail sur le plan théorique et explicitant les concepts auxquels nous nous

référons au cours d'une revue de littérature, nous établissons ensuite le cahier des charges de notre logiciel, et nous explicitons notre méthodologie de développement. Dans un quatrième temps, nous décrivons l'implémentation du programme et nous le soumettons à une évaluation préliminaire. Enfin, nous discutons la portée de notre travail et présentons nos conclusions.

2. Revue de littérature

2.1 Les réseaux sociaux

Notre travail s'inscrit à l'interface de plusieurs courants de recherche. Il fait notamment référence à la notion de réseau social. Un réseau social est une carte de relations entre des individus, indiquant la manière dont ils sont connectés au travers de degrés de proximité sociale variés allant de la connaissance fortuite à des liens familiaux rapprochés (Wikipedia, notre traduction). Le terme de réseau social est un terme de BARNES datant des prémisses de la sociométrie et des sociogrammes, au début 1954.

De nombreuses disciplines s'intéressent aux réseaux sociaux, telles les sciences sociales ou la psychologie organisationnelle, pour leurs implications humaines, stratégiques, ou encore politiques. Si la théorie des réseaux sociaux (*"Social Network Theory"*) s'attache à mettre en lumière les effets de la structure sociale des relations d'une personne, d'un groupe ou d'une organisation sur ses croyances et ses comportements, l'analyse de réseaux sociaux (*"Social Network Analysis"*) propose des méthodes pour détecter et mesurer l'ampleur des pressions causales inhérentes à cette structure sociale.

Ainsi, la réalité doit être envisagée sous le jour des propriétés des relations entre personnes ou organisations, et non pas sous le jour des propriétés des entités en question. Chaque individu est un noeud connecté aux autres par le biais de liens de diverses sortes. Chacun dispose d'un *"capital social"* (PUTNAM, 1995) caractérisé par la valeur collective de tous les réseaux sociaux, et par leur susceptibilité d'agir les uns pour les autres. Certains l'envisagent comme un investissement actif dans l'espoir d'un retour sur le marché économique (LIN, 2001), ou encore comme la capacité à maintenir un réseau durable de relations institutionalisées d'acquaintance et de reconnaissance mutuelle (BOURDIEU, 1986).

Le pouvoir correspond dans cette optique à une position nodale dans un réseau de relations. Certains individus se détachent par leur capital social : on parle dans ces cas de leaders d'opinion. Ces personnes ont des rôles cruciaux pour la diffusion des idées et des innovations. En effet, les réseaux sont parcourus par des flux de communication (*"Communication Networks")* pouvant être définis comme les *"liens interpersonnels créés par le partage d'information au sein du réseau"* (ROGER, 1986, notre traduction).

De nombreuses applications ont vu le jour dans le but de soutenir les interactions de groupe. La cote de popularité de ces applications est arrivée à un apogée ces dernières années, mais leur histoire est bien plus

longue, comme le montre ALLEN (2004), qui s'est attaché à retracer l'évolution des différentes terminologies. En effet, la notion programme social ("*Social Software*"), et l'idée même d'utiliser des ordinateurs pour collaborer remonte à 1945, avec *"Memex"*, l'ancêtre du PC, inventé par VANNEVAR BUSH. Puis il faut attendre jusqu'en 1960, pour que l'idée émerge à nouveau et aboutisse à la création de l'"*Advanced Research Projects Agency"* (ARPA) en réponse au lancement du satellite Spoutnik par les Russes. ALLEN rappelle les jalons de l'histoire des programmes sociaux, de la naissance de l'*Electronic Information Exchange System* » (EIES) en 1974 : un système de téléconférence conçu pour que les groupes « *agissent avec leur intelligence collective plutôt que leur plus petit dénominateur commun* » à l'émergence du terme « g*roupware* » en tant que *« processus des groupes intentionnels soutenus par un programme informatique » (*JOHNSON – LENZ, 1978).

En 1984, GREIF et CASHMAN inventent le terme de «*travail collaboratif assisté par ordinateur* », Supported Collaborative (parfois Cooperative) Work (CSCW) en version originale. Le terme qualifierait le champ d'étude de la technologie utilisée pour collaborer et ses implications psychologiques, sociales et organisationnelles, par opposition à celui de « groupware » qui désignerait plus particulièrement la technologie informatique sous-jacente. Le premier terme finira par tomber dans l'oubli alors que le second sera repris par les multinationales pour désigner par extension tout programme qui supporte une utilisation par des utilisateurs multiples. Le terme de « *programme social* » *(« Social Software »)* commence à être à la mode dès le début des années 90 mais reste peu usité en dehors des milieux spécialisés. Les efforts de SHIRKY (2002) contribuent à sa vulgarisation. SHIRKY définit actuellement les programmes sociaux comme des *« programmes qui supportent les interactions de groupe ».*

Un très grand nombre d'applications de «*SoSoWa* » ont vu le jour récemment (Friendster, Orkut. Tribe.net)… Les fonctionnalités proposées communément sont le support pour les interactions conversationnelles entre personnes ou groupes (messagerie instantanées, espaces collaboratifs virtuels), le support pour la rétraction sociale (système de réputation et d'estimation de la crédibilité) et le support aux réseaux sociaux (représentation numérique du réseau social et facilitation de l'ajout de connexion (KAPLAN-LEISERSON, 2003). Certains logiciels favorisent certains de ces aspects comme la collaboration à distance (Groove, collaboration peer to peer), ou la gestion des connaissances (AskMe, proposition de définitions et de documents pertinents, ActiveNet, recherche continue au sujet des intérêts de l'utilisateur).

On assiste à un détournement commercial du phénomène des réseaux sociaux, transformant les « *SNS* » (« *Social Network Software* ») en « YASN » (« *Yet Another Social Software* »), littéralement : « *encore un de ces programmes sociaux* ». Il devient difficile de saisir le point de mire de certains programmes, tant le foisonnement des applications est impressionnant. La Wikipedia en mentionne plus de 200. Le fonctionnement est souvent similaire, un certain nombre de membres fondateurs envoie des messages d'invitations à leur réseau personnel pour qu'ils rejoignent le site, les nouveaux membres répètent le procédé, constituant un réseau de relations. Les sites proposent des fonctionnalités comme la mise à jour de carnet d'adresse, les profils visualisables, la capacités de former de nouveaux liens par des services

d'introduction, et autres formes de connexion sociale. Les réseaux peuvent être organisés autour de contacts professionnels, comme c'est le cas pour Ecademy ou LinkedIn. Friendster occupe la tête de liste avec 6 millions d'utilisateurs connectés. Friendster a vu se développer des profils d'individus fictifs faisant référence à des célébrités, les « fakesters ». Depuis février 2005, le site propose un service de weblogs que les utilisateurs peuvent connecter à leurs profils. Des stars bien réelles possèdent même un blog Friendster.

Friend-Of-A-Friend est une initiative hybride entre les programmes sociaux et le Web sémantique. Comme nous l'avons mentionné dans un article préalable (GENET, 2004) : *« FOAF est une tentative de construire un réseau de fichiers RDF (Ressource Description Framework), qui décrirait des personnes réelles sur le web réel. Il est issu d'un effort communautaire pour exprimer des métadonnées sur les personnes, leurs intérêts, leurs relations et leurs activités. Il fait partie d'une initiative plus vaste dans le domaine du Web sémantique, qui vise à créer un Web dont les données pourraient être traitées par des machines. (…) FOAF permet de récolter et de réunir de très grandes quantités de données extrêmement rapidement. Le vocabulaire FOAF consiste en une collection de définitions RDF mise à jour par les développeurs de l'organisation RDFweb. »* FOAF permet de présenter une personne et son réseau social dans un langage lisible par les machine. On peut intégrer ces données dans des pages Web ou dans des programmes sociaux (Academy, Plink…). Le réseau constitué par les profils FOAF publiés sur Internet est aussi un véritable réseau social, directement exploitable par les machines. A la différence des données générées par les programmes sociaux usuels, les données des profils FOAF sont contrôlées par leurs propriétaires, comme celles des pages personnelles.

Voici un exemple de code FOAF :

```
<RDF:RDF xmlns:RDF=http://www.w3.org/1999/02/22-RDF-syntax-ns#
xmlns:RDFs=http://www.w3.org/2000/01/RDF-schema#
xmlns:foaf="http://xmlns.com/foaf/0.1/"xmlns:geo=http://www.w3.org/2003/01/geo/wgs84_pos#
xmlns:dc="http://purl.org/dc/elements/1.1/">
<foaf:Person>
<foaf:name>Nom complet</foaf:name>
<foaf:title>Titre</foaf:title>
<foaf:firstName>Prénom</foaf:firstName>
<foaf:surname>Surnom ou nom de famille</foaf:surname>
<foaf:mbox RDF:resource="mailto:e-mail"/>
<foaf:mbox_sha1sum> sum URI SHA1 de votre e-mail</foaf:mbox_sha1sum here>
<foaf:homepage RDF:resource="URL de votre homepage"/>
<foaf:depiction><foaf:Image RDF:about="URL de votre photo"/></foaf:depiction>
<foaf:gender>Sexe</foaf:gender>
<foaf:icqChatID>ID ICQ</foaf:icqChatID>
<foaf:aimChatID>ID pour les chats AIM</foaf:aimChatID>
```

<foaf:workplaceHomepage RDF:resource="URL de votre lieu de travail"/> <foaf:based_near> <geo:Point geo:lat="Latitude géographique de l'endroit où vous vivez" geo:long="Longitude géographique de l'endroit où vous vivez"/>
</foaf:based_near>
<foaf:made RDF:resource="URI/URL d'un document que vous avez fait" />
<foaf:interest RDF:resource="URL de votre domaine d'intérêt" dc:title="Domaine d'intérêt" />
</foaf:Person>
</RDF:RDF>

2.2 La cartographie sociale

L'analyse de réseaux sociaux est issue d'une discipline descriptive et analytique, mais les images ont rapidement pris une place importante dans ce domaine. Selon FREEMAN (2000), les images de réseaux sociaux ont fourni aux chercheurs des indications sur les structures des réseaux, et les ont aidé à les communiquer aux autres. Depuis les premiers sociogrammes de MORENO (1932), de nombreuses représentations graphiques ont vu le jour, certaines accentuant les traits de la structure du groupe, d'autres les similarités et les différences entre les positions occupées par les acteurs, ou les deux à la fois. Pour MORENO (1953), la représentation visuelle est le seul moyen d'analyse structurel de la communauté.

Différents types de représentations se sont succédés historiquement. MORENO (Ibid, 1953) a imaginé plusieurs modes de représentations en nuages de points mettant en avant les caractéristiques des acteurs et de leurs relations. LUNDBERG et STEELE (1938) sont à l'origine de la variation de la taille des points pour montrer les différences de statuts sociométriques, et NORTHWAY (1952) du graphique en forme de cible. On doit les premières procédures computationnelles de positionnement de points à l'aide de l'analyse factorielle à BOCK et HUSAIN (1952), et PROCTOR (1953). Le premier programme informatique de production d'images est développé au début des années 1970 (ALBA, 1972), il positionne les points à l'aide d'un traitement statistique appelé multi-dimensional scaling. En 1996, BATAGELI et MVAR donnent naissance au premier programme complet d'analyse de réseaux sociaux : *Pajek*, suivi de près par *NetVis* de KREMPEL, qui permet de localiser des données en deux modes, et *Multinet* (de RICHARDS et SEARY) qui donne l'illusion de la 3D. Enfin, de nombreuses applications ont été développées récemment avec la révolution des navigateurs Web, le plus souvent à l'aide des technologies JAVA et VRML : *InFlow, TouchGraph, Social Network Visualiser, Vizster, Pajek, Ucinet, Lapack, Giny, JUNG, Hypergraph, JDigraph, Wilmascope, IVC, JGraphEd, VGJ, Zoomgraph, Prefuse, Walrus, ZVTM, Infovis Toolkit, Large Graph Layout…*

Certains scientifiques se sont attachés à développer des applications de visualisations de réseaux sociaux et à les intégrer dans des environnements d'apprentissage. DOBSON et al. (2004) ont utilisé la visualisation de réseaux sociaux pour montrer le flux temporel des échanges de communications dans un environnement d'apprentissage. Ils concluent que les visualisations permettent de trouver une personne spécifique pour accomplir une tâche spécifique et qu'ils peuvent suggérer des personnes précises pour la collaboration dans les communautés d'apprentissage. Dans cette perspective, l'illustration des liens

fondant une communauté sociale peut se révéler utile sur le plan pédagogique. La localisation des ressources pertinentes est un des enjeux de la gestion des connaissances, ou « *knowledge management* » en anglais. Ce courant de recherche a grandement contribué à l'alimentation de nos réflexions et à l'orientation de notre travail.

2.3 La gestion des connaissances et les systèmes d'apprentissage distribués

Il est souvent ardu de trouver la bonne personne au sein d'un groupe ou d'une organisation, lorsque l'on cherche un interlocuteur pour lui poser une question. En règle générale, trouver la personne adaptée prend davantage de temps que le temps nécessaire pour répondre à la question. Une nouvelle génération de programme, les systèmes de gestion des experts, automatise ce processus de localisation de personnes et de documents pouvant répondre à une question donnée. Contrairement aux systèmes de gestion des connaissances qui stockent l'information et laissent les utilisateurs le soin d'y chercher ce dont ils ont besoin, ces programmes utilisent des technologies sophistiquées pour trier les documents et e-mails déjà stockés et pointer vers des gens et des contenus qui seront les plus utiles. Les informations sont évaluées en termes de fréquence d'utilisation, d'échelle temporelle, ou de degré d'utilité pour les utilisateurs précédents, et le programme dirige les personnes en besoin d'information vers ceux qui peuvent les aider. Les experts peuvent être contactés par e-mail, téléphone, ou en personne.

Selon KAPLAN-LEISERSON (op.cit. 2003), les systèmes de localisation d'experts peuvent améliorer le flux des connaissances et de l'information, aider à déceler les leaders de pensée et ceux qui détiennent les informations clés, et cibler les opportunités où un flux de connaissance amélioré aura le plus d'impact en fonction des objectifs à atteindre.

SOLLER, GUIZZARDI, MOLINI et PERINI (2004) ont cherché à améliorer l'apprentissage organisationnel distribué et la gestion des connaissances. Ils sont partis du constat que si les connaissances étaient partagées de manière effective dans les organisations comportant des puits d'expertise spécialisées, les organisations apprenaient et se développaient mieux, et leur productivité et leur expansion étaient améliorées. En effet, les organisations partagent naturellement les connaissances en formant de petits groupes basés sur les intérêts similaires, les affinités personnels et la confiance. Ces groupes sont appelés *« communautés de pratique »* et fonctionnent comme des entités cohésives qui partagent buts et centres d'intérêts. Le terme dérive des travaux de LAVE et WENGER (1991). A la suite de WENGER et al. (1998), de nombreux chercheurs envisagent la promotion des communautés de pratiques comme une stratégie de développement et de gestion des savoirs. Ces communautés facilitent le partage et la création de nouvelles connaissances et contribuent à la stabilité et au développement des organisations. L'apprentissage est ainsi essentiellement ancré dans des pratiques collectives.

SOLLER et al. ont développé deux outils visant à améliorer un environnement d'apprentissage organisationnel : un outil de visualisation de réseaux sociaux, et un agent de recommandation conscient des implications sociales. Selon SOLLER et al., dans une organisation distribuée, l'expertise existe, mais de manière dispersée dans les compréhensions (comprises intérieurement et partagées) de leurs connaissances respectives par chacun des membres et dans les similarités comportementales et cognitives (cachées) entre individus. C'est pourquoi un environnement d'apprentissage distribué doit aider à mettre en lumière et rendre saillantes ces caractéristiques cachées afin que les apprenants prennent conscience des connaissances organisationnelles et des communautés de pratiques correspondantes. Les auteurs ont utilisé une méthode d'analyse : l'analyse intentionnelle, afin de mettre à jour les facteurs critiques qui influencent le partage des connaissances et l'apprentissage et ont pu réaliser des variables de modèles d'utilisateurs, qui ont fondé leur système de recommandation. Ils ont amélioré l'interface à l'aide d'une visualisation de réseau social interactive qui affiche les résultats à chaque requête en termes de facteurs sociaux et comportementaux, et qui promeut la perception des communautés de connaissance en ligne. Mettre en lumière les caractéristiques particulières des uns et des autres permet la localisation des connaissances et contribue à une perception plus transparente du fonctionnement organisationnel, mais quelles sources d'informations sur les gens peut-on exploiter pour ce faire ?

2.4 L'analyse de pages personnelles

Les chercheurs en sciences sociales récoltent le plus souvent leurs données expérimentales à l'aide de méthodes fastidieuses telles que les entretiens téléphoniques ou les entrevues en tête à tête. Internet fournit pourtant des quantités très importantes d'informations sur les gens, leurs intérêts et leurs expériences. LADA ADAMIC XEROX (2001), soulignent que l'une des applications principales du Web est de diffuser des pages personnelles. *« Ces pages le plus souvent autobiographiques contiennent toutes sortes de données allant de la photo de l'animal de compagnie à l'essai littéraire ou au curriculum vitae. Les pages personnelles ne flottent pas de manière isolée sur le Web mais pointent vers d'autres pages et sont pointées par d'autres utilisateurs, de véritables amis et voisins du Web. Ces liens peuvent prendre toutes sortes de significations, de l'amitié à la collaboration, à l'intérêt général pour le matériel contenu dans la page personnelle d'un autre utilisateur. De cette manière les pages personnelles deviennent une part de la structure de la communauté à grande échelle.»* (notre traduction).

Pour certains auteurs, la topologie des liens entre pages Web permet de faire émerger des communautés virtuelles constituées de pages traitant des mêmes sujets (FLAKE et al, 2000. GIBSON et al., 1998, LARSON, 1996). Ainsi, il est possible d'extraire de larges réseaux sociaux de pages personnelles d'individus à priori isolés. Les données peuvent être extraites de plusieurs sources : le texte des pages personnelles qui fournit des indications sémantiques sur le contenu de la page, la co-occurrence de termes, les liens pointant vers l'extérieur (« *out-links* »), vers l'intérieur (« *in-links* »), ou encore les listes de courrier « *mailing lists* » (LADA ADAMIC XEROX, op. cit. 2001). Les mots publiés par les personnes et les structures de leurs pages personnelles permettent donc de faire émerger des caractéristiques intra- et inter-personnnelles. Les données sont facilement traitables puisqu'elles existent

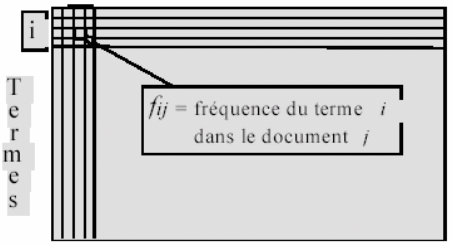
déjà sous forme numérique. Cependant, des outils de recherche d'informations et d'indexation sont nécessaires pour traiter l'information en grande quantité. Si la topologie des pages Web permet de faire des inférences probabilistes quant à leur signification, certaines techniques statistiques d'indexation tentent de faire émerger le sens des contenus textuels à partir des contextes d'occurrence des termes. C'est le cas des techniques d'indexation par analyse de la sémantique latente.

2.5 Les techniques d'indexation par analyse de la sémantique latente

La recherche et l'indexation de documents en rapport avec un sujet donné peuvent se révéler extrêmement ardues dans la mesure où le sens des concepts est interprété de manière divergente en fonction de chaque individu. La polysémie et la synonymie qui caractérisent notre langage complexifient encore la tâche. L'indexation par analyse latente (« *Latent Semantic Analysis* ») est une méthode de recherche d'information qui tente de pallier ces difficultés, en faisant appel à une technique statistique, la décomposition en valeurs singulières (SVD, « *Singular Value Decomposition* »). Cette méthode permet de faire émerger des structures sémantiques latentes au sein des documents, en constituant un espace sémantique de grande dimension à partir de l'analyse statistique de l'ensemble des co-occurrences dans un corpus de textes. Elle se sert d'une matrice de données associant termes et documents et construit un espace à l'intérieur duquel termes et documents sont associés étroitement et placés les uns à côtés des autres (DEERWESTER et al., 1990). La SVD permet la réorganisation de cet espace afin de refléter que les principales structures associatives des données grâce à la réduction du nombre de dimensions considérées. De cette manière, des termes qui n'apparaissaient pas dans un document peuvent être considérés comme proches de ce document s'ils sont consistants avec les structures associatives émergentes des données. La position dans l'espace fait office de technique d'indexation sémantique inédite : les informations sont trouvées grâce à des termes composant une requête pour identifier un point dans l'espace, et les documents dans ses environs sont retournés. BESTGEN (2004), a repris les différentes étapes nécessaires pour dériver un espace sémantique d'un tableau lexical. Nous reproduisons ses illustrations ci-après.

Annexe 1 : Les étapes d'une analyse sémantique latente

1) Obtention d'un tableau lexical
« termes * documents »
(nombre d'occurrences de chaque
terme dans chaque document)



2) Transformation des fréquences
afin de privilégier les termes
les plus informatifs

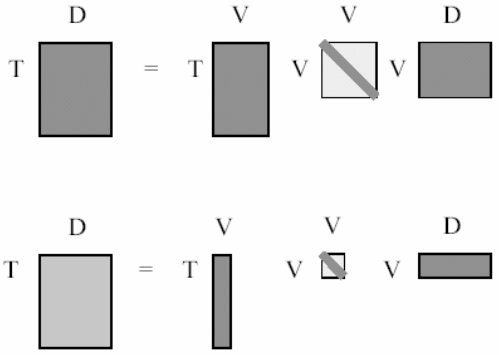
Transformation des fréquences
afin de privilégier les mots les plus informatifs

$$f_{ij}' = \frac{\log(f_{ij}+1)}{-\sum_j \frac{f_{ij}}{\sum_j f_{ij}} \log\left(\frac{f_{ij}}{\sum_j f_{ij}}\right)}$$

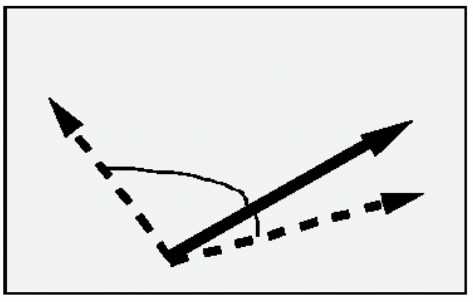
Pondération locale :
réduit l'impact des mots
très fréquents dans un document

Pondération globale :
réduit l'impact des mots « peu informatifs », c'est-à-dire qui
apparaissent avec une fréquence constante dans les documents
Exemples : mots peu informatifs : faire, pouvoir, deux, nouveau
mots très informatifs : neutrino, cancérogène, fiacre

3) Décomposition en valeurs singulières
- Compression de l'information par la sélection des k dimensions orthogonales les + importantes ($100 \leq k \leq 300$)
- Permet d'obtenir les vecteurs qui représentent les termes dans l'espace comprimé



4) Emploi :
- Calculer la proximité sémantique entre des mots ou des segments
- Le sens d'un mot est représenté par un vecteur
- La similarité entre deux mots est mesurée par le cosinus entre les vecteurs correspondants (idem pour les segments)



Y. BEGSTEN (Annexe au texte : « Analyse sémantique latente et segmentation automatique des textes »)

La méthode de l'analyse de sémantique latente se sert de données sur la co-occurrence contextuelle naturelle des mots (LANDAUER, LAHAM, 1998). Les fréquences avec lesquelles certains mots apparaissent, en tant que composant de productions langagières naturelles, permettent d'inférer des données associationnelles. Les régularités statistiques reflétées sont des relations entre des expressions de sens similaires. Les contraintes mutuelles inhérentes à une multitude de relations de co-occurrences sont satisfaites en étant forcées en une représentation globale de dimensionnalité moins importante, grâce à la décomposition en valeurs singulières.

La décomposition en valeurs singulières est aussi appelée analyse factorielle à deux modes. Elle part d'une matrice rectangulaire arbitraire, constituée de lignes et de colonnes composées par différentes entités, par exemple des termes et des documents. Cette matrice est ensuite décomposée en 3 autres matrices grâce à la SVD, donnant lieu à des « vecteurs » ou « valeurs singulières ». Certaines de ces valeurs, de moindre importance, vont être abandonnées pour constituer un modèle approximatif qui contient des dimensions moins nombreuses. Au sein de ce modèle réduit, toutes les similarités entre termes, documents ou termes et documents, sont approximées par des valeurs quant à ce nombre de dimensions restreint. Les résultats peuvent être représentés géométriquement dans une configuration spatiale dans laquelle les produits à point (« dot product ») ou les cosinus entre les vecteurs représentant deux objets correspondent à la réalité.

L'analyse de sémantique latente n'est pas capable de capturer le sens complet d'un texte. Tout d'abord parce que le corpus langagier utilisé pour générer l'espace sémantique de référence ne sera jamais représentatif de l'ensemble des mots auquel un humain est confronté. Deuxièmement parce que la LSA ne reflète pas les relations d'ordre entre les mots et ne tient pas compte de la syntaxe. Dans la mesure où l'analyse ne reflète que des similarités de continuités, elle ne peut pas être utilisée pour induire certaines classes de relations structurelles, dont les opérations booléennes ou les relations causales. Plusieurs scientifiques se sont attachés à estimer les facultés de l'analyse de sémantique latente en termes de simulation du jugement humain. LANDAUER et DUMAIS (1997), l'on entraînée sur une encyclopédie pour lycéens (environ 5 millions de mots) puis l'ont soumise au test de détection des synonymes du TOEFL (« Test Of English as a Foreign Language »). L'algorithme s'est révélé efficace dans 64 % des cas. Il apparaît que l'analyse de sémantique latente peut simuler la cognition humaine dans de nombreux cas, probablement parce que le jugement humain se base souvent plus sur les relations de similarités que sur la logique syntaxique discrète (TVERSKY et KAHNEMAN, 1983). La LSA a été utilisées dans plusieurs expériences de recherches d'information, de traduction, de comparaison de textes produits par des étudiants, ou encore de comparaison de sens. Elle présente de nombreuses opportunités, notamment en intelligence artificielle pour les systèmes apprenants, et constitue en outre un modèle de compréhension de la logique humaine intéressant pour les sciences cognitives.

Au cours de cette revue de littérature, nous avons envisagé successivement les apports de la théorie des réseaux sociaux et de la carthographie sociale, de la gestion des savoirs, de l'analyse de pages personnelles et des techniques d'indexation par analyse de sémantique latente. De la théorie des réseaux

sociaux et de la carthographie sociale, nous avons retenu l'impact de la représentation des réseaux sociaux pour la prise de consience de la localisation des ressources. La gestion des savoirs nous a permis de mesurer l'importance de la promotion et de la mise en évidence de « communautés de pratique » pour favoriser la diffusion des connaissances. L'analyse de pages personnelles a suggéré des possibilités d'extraction de communautés en ligne à partir de quantités de données importantes contenues dans des pages personnelles. Enfin, nous avons entrevu les capacités de l'analyse de sémantique latente pour dépasser les limites des techniques d'indexation usuelles et détecter des synonymes lexicaux et des similarités entre documents. Notre projet a été alimenté par ces différentes perspectives. Nous avons imaginé un système de support à la diffusion des connaissances qui permette de montrer des liens inter-personnels, de soutenir les interactions de groupe, de favoriser les contacts et la collaboration. Nous souhaitions générer des métadonnées sur les personnes et les utiliser pour représenter des communautés de pratiques. Nous désirions que le système extraie et résume les informations contenues dans les pages personnelles pour améliorer la perception des communautés de connaissance en ligne. Afin d'élaborer un tel prototype, nous nous sommes attachés à définir les spécifications fonctionnelles de notre logiciel. Nous les reprenons dans le chapitre suivant.

3. Cahier des charges du logiciel

Nous nous avions imaginé un logiciel composé de trois modules, dont les objectifs principaux pouvaient être définis comme suit :

-Générer automatiquement des profils de personnes à l'aide des informations contenues dans les pages personnelles stockées sur les serveurs du TECFA, et retranscrire ces profils en langage machine à l'aide du vocabulaire Friend-OF-A-Friend.
-Répondre à des requêtes textuelles sur les profils des gens et permettre la comparaison entre les centres d'intérêts et les réseaux relationnels de chacun.
-Permettre de visualiser certaines informations extraites des profils des gens.

3.1 Cahier des charges détaillé du module 1 : Générateur de profils Friend-Of-A-Friend

Nous avions songé à élaborer un logiciel qui permette de résumer les informations contenues dans les pages personnelles des membres du TECFA. Afin d'exploiter un maximum d'informations, le module devait utiliser aussi bien les données structurées (annuaire LDAP, données XML), que les données non structurées (texte contenu dans les pages). Le langage FOAF, parce qu'il permet de « *créer des pages Web lisibles par les machines décrivant les gens, leurs liens, et les choses qu'ils créent et font* » (Wiki foaf), nous était apparu comme une solution particulièrement pertinente de synthèse des informations extraites des différentes sources de données. Notre programme devait effectuer une conversion des

données extraites des pages personnelles vers le langage FOAF. Dans un premier temps, nous avions formulé sommairement les exigences quant aux possibilités du logiciel. Certaines idées sont venues augmenter à posteriori ce cahier des charges initial.

3.1.1 Exploitation des données du serveur annuaire LDAP

Lorsque l'on cherche à extraire des informations à partir de productions humaines spontanées, il est nécessaire d'établir les caractéristiques distinctives de ce que l'on cherche. De nombreux algorithmes de traitement du langage naturel cherchent ainsi des régularités schématiques dans les données qu'ils parcourent. Ce type de recherche n'est plus possible lorsque l'on cherche un nom commun ou l'adresse d'une page personnelle, car les possibilités de définition de la cible se déclinent à l'infini. Une source non négligeable de données structurées et régulièrement mises à jour se trouvaient à notre disposition : les annuaires LDAP de l'Université de Genève, et plus particulièrement ceux du TECFA. Ces données pouvaient nous permettre de contourner les difficultés de recherche de certaines informations. Notre programme devait donc être capable de se mettre en relation avec le serveur LDAP et d'en extraire les informations suivantes : prénom, nom de famille, adresse e-mail, adresse de la page « travaux » (pages portfolio publiées par les étudiants, sur lesquelles ils présentent leurs travaux), pages personnelles à l'université, et groupes d'appartenance (étudiants, enseignants ou personnel).

3.1.2 Extraction des données des pages personnelles

Si certaines données pouvaient facilement être extraites de sources de données structurées comme celles contenues dans les annuaires LDAP de l'université, une grande partie des informations existait sous forme purement textuelle. C'est pourquoi notre programme devait identifier toutes les productions textuelles d'une personne. Cela sous-tendait qu'il soit capable de différencier la production d'une personne par rapport à celle d'une autre, et de ne répertorier qu'un seul exemplaire de chaque page produite.

Le programme devait être capable d'identifier le contenu d'une page Web comme étant la production d'une personne précise, et les liens dans cette page pointant vers d'autres pages internes au site de cette même personne. Par opposition, il devait aussi pouvoir identifier et stocker des liens pointant vers des pages externes. Le stockage des « *out-links* » (liens vers l'extérieur) permettrait éventuellement d'effectuer une analyse topologique des pages.

Nous disposions de plusieurs pistes pour tenter d'exploiter les données textuelles produites par les gens et d'en faire émerger des centres intérêts saillants, dont l'analyse lexicométrique du texte, et l'indexation à l'aide d'un algorithme utilisant l'analyse de sémantique latente. Afin de soumettre le texte à ces techniques, il était nécessaire de regrouper les différentes pages produites par une personne sous forme de

document unique. Le programme devait donc générer un fichier texte à l'aide de toutes les pages produites par la personne.

Constituer un corpus textuel ne se résume pas à une simple copie du texte contenu dans une ou plusieurs pages Web dans un fichier commun. Le contenu d'informations non structurées contenues dans les pages Web n'est pas du simple texte lorsqu'on l'importe depuis une interface de programmation comme php. En effet, à ce stade il s'agit encore de code HTML, composé de balises qu'il nous faudra éliminer avant pouvoir accéder à du texte pur. Conserver les balises HTML confondues avec le texte et les analyser sous forme textuelle pourrait conduire à un biais, notamment du fait de la répétition de la structure HTML dans chacune des pages. En outre, il existe des parasites d'un autre type qu'il nous fallait neutraliser : les mots vides. « *Les mots vides sont des mots ignorés lors d'une requête dans les outils de recherche, car leur utilisation n'améliore en rien la pertinence des résultats, dans la mesure où ils sont trop souvent utilisés. (…) Trop communs pour être interrogeables, ils ne sont pas indexés par les outils de recherche parce qu'ils créent de fausses occurrences.* » (Définition de l'Office Québécois de la langue française). Notre programme devait supprimer les balises HTML et les mots vides contenus dans une liste de mot vides.

Certains algorithmes d'indexation de documents comptent les occurrences de mots et ne retiennent que les mots utilisés à plusieurs reprises comme significatifs. Une technique d'analyse lexicométrique de base consiste à évaluer la portée du vocabulaire d'un individu, ou encore les mots qu'il utilise le plus fréquemment. Notre programme devait compter les mots utilisés par une personne, leurs occurrences multiples, et trier les mots les plus utilisés.

Une autre source de données structurées était exploitable sous la forme des pages XML publiées par les étudiants. Ces pages présentant les travaux des étudiants contenaient des liens vers leurs productions personnelles, mais aussi des données sur leur pseudonyme courant dans les applications en ligne (telles que le Moo du TECFA), ou encore leur identifiant sur les ordinateurs de l'université (login UNIX). Notre programme devait extraire ces données des balises XML correspondantes.

La seule analyse topologique des « out-links » figurant sur une page personnelle nous paraissait insuffisante pour situer clairement un individu au sein de son réseau social. C'est pourquoi nous souhaitions que notre logiciel soit capable de reconnaître le nom d'un collègue de travail ou d'un autre étudiant mentionné dans le texte de la page personnelle, ou dans les données de la page de présentation des travaux de la personne. Une fois cette reconnaissance effectuée, le logiciel devait aussi effectuer une requête sur les informations disponibles dans le profil de cette autre personne, et récolter transitivement son nom, prénom et adresse e-mail.

3.1.3 Génération des fichiers Friend-Of-A-Friend

Le logiciel devait dans un dernier temps convertir les différentes données dans un langage lisible par les machines et facilement manipulable : FOAF. Il devait utiliser les différentes informations recueillies pour les stocker dans des balises FOAF correspondant le mieux possible. Les variables nom, prénom, e-mail, page personnelle, page travaux et groupe devaient être intégrées aux balises FOAF suivantes : *foaf :givenname, foaf :family_name, foaf :name, foaf:mbox, foaf:homepagefoaf:workplaceHomepage*. Les variables issues du XML devaient être intégrées au balises *foaf:nick* (pseudonyme) et *foaf:OnlineChatAccount* (login UNIX), et *foaf:made* (productions personnelles). Quant aux mots utilisés fréquemment, ils devaient être stockés dans des balises *foaf:topic_interest*. Les « amis » mentionnés sur les pages personnelles et les informations leurs correspondant seraient stockées dans des balises imbriquées successivement *foaf :knows, foaf :Person, foaf :name* …et ainsi de suite.

Ce procédé, basé sur le postulat que des mots utilisés fréquemment par une personne correspondent à ses centres d'intérêts, est par ailleurs tout à fait discutable, mais nous reviendrons sur ce débat dans la partie de ce texte réservée à l'autocritique.

Pour comparer les données et fournir un point de départ centralisé pour l'analyse des fichiers FOAF, il était nécessaire de créer un fichier listant et reliant les fichiers FOAF, et de le tenir à jour. Le programme devait générer ce fichier à chaque nouvelle génération de profil.

Les navigateurs Web affichent le RDF comme du XML, sous forme de structure arborescente. La mise en forme en est absente. Nous désirions que notre programme permette un affichage visuel acceptable du profil FOAF généré.

3.1.4 Implémentation d'un algorithme d'indexation par analyse de sémantique latente

Une autre possibilité de mise en évidence de mot clés significatifs pour mettre en lumière les centres d'intérêts d'une personne consistait en l'usage d'un algorithme d'analyse de sémantique latente. Nous désirions que notre logiciel implémente un tel algorithme. Nous avons finalement renoncé à son utilisation dans ce module du logiciel, pour lui préférer une simple indexation basée sur la poly-occurrence. Nous reviendrons plus loin sur les raisons ayant motivé ce choix. La LSA devant être entraînée sur un corpus de référence, le programme devait générer un fichier corpus général regroupant tous les corpus et le mettre à jour.

3.2 Cahier des charges détaillé du module 2 : Interrogateur de profils Friend-Of-A-Friend

Nous souhaitions élaborer un module qui nous permette d'interroger les fichiers FOAF générés à l'aide du module 1. Nous voulions comparer d'une part les réseaux sociaux des personnes, et d'autre part leurs centres d'intérêts. Les personnes partageant des intérêts communs et présentant des relations, bidirectionnelles ou non, devaient être repérables. Le programme devait comporter une interface qui permette d'entrer des requêtes sur les différentes balises RDF présentes dans les fichiers FOAF. Le logiciel devait offrir une possibilité de recherche des personnes selon leurs centres d'intérêt : entrer un mot clé dans un champ et renvoyer les personnes intéressées par ce mot, en fonction du contenu des balises *foaf:topic_interest* contenues dans les fichiers FOAF. Le logiciel devait, si possible, implémenter l'algorithme LSA pour effectuer une requête sur le mot clé et ainsi que sur les mots les plus proches sur le plan sémantique (synonymes) de ce mot. Afin de permettre de contacter facilement une personne pouvant servir de ressource d'apprentissage, ou d'échanger au sujet de centres d'intérêts communs.

Savoir qui partage notre intérêt pour une chose donnée peut être d'une aide précieuse, on peut néanmoins souhaiter en savoir plus sur les préoccupations d'une personne dont on ne sait rien. C'est pourquoi notre programme devait offrir la possibilité d'entrer le nom ou le prénom d'une personne et d'obtenir en réponse ses centres d'intérêts.

Le module 1 (générateur de FOAF) devait identifier les personnes mentionnées dans le texte produit dans les pages personnelles. Nous souhaitions donc que le module 2 permette d'interroger cet aspect (balise *foaf:knows*). Le programme devait donc montrer les personnes connues par quelqu'un, et mettre à disposition certaines information les concernant : nom, prénom et adresse e-mail.

La bidirectionnalité des relations entre deux personnes peut être un indicateur d'un degré de proximité plus élevé, ou d'une reconnaissance mutuelle. Notre logiciel devait permettre d'interroger les « in-links » relationnels de quelqu'un : les personnes qui le ou la connaissent.

3.3 Cahier des charges détaillé du module 3 : Visualiseur de profils Friend-Of-A-Friend

Le dernier volet de notre environnement devait offrir une visualisation des données contenues dans les profils FOAF. Il devait comporter un champ pour entrer le nom ou le prénom d'une personne, et représenter en retour les liens sociaux mentionnés par celle-ci. Notre cahier des charges prévoyait d'afficher une représentation globale de tous les protagonistes du réseau social composant l'annuaire LDAP de l'université, et de leurs relations. Nous souhaitions aussi représenter les liens vers l'extérieur

afin de montre la topologie d'une page personnelle à l'échelle du Web. Nous n'avons pu terminer le développement de cette fonctionnalité faute de temps.

L'analyse de sémantique latente permet de comparer des documents, nous souhaitions montrer les degrés de similarités entre les productions textuelles constituant les pages personnelles de chacun.

Dans un souci de promotion des interactions interpersonnelles, nous souhaitions que notre programme offre des fonctionnalités d'affichage rapide des informations relatives à une personne, et permette d'accéder sa page personnelle facilement.

Une fois avoir formulé les exigences auxquelles notre prototype se devait de répondre, le travail de développement pouvait commencer. Nous explicitons nos méthodes de travail dans le chapitre qui suit.

4. Méthodologie de développement

VAN AKEN a défini la visée de la « science-conception » comme « le développement de connaissances sur la conception et la réalisation de quelque chose, par exemple de résoudre des problèmes de construction, ou l'amélioration des performances d'entités existantes, en d'autres termes pour implémenter des innovations » (VAN AKEN, 2004, notre traduction). Elle concerne trois sortes de processus de conception : design orienté objet, design pour l'intervention ou pour l'artefact (design de réalisation : implémentation de l'intervention ou construction de l'artefact) et design procédural (cycle de résolution de problèmes ou méthode pour concevoir la solution des problèmes). Elle fournit des prescriptions générales qui interprétables en fonction de problèmes spécifiques. Ces « *règles technologiques* » (BUNGE, 1967) sont des connaissances liant un objet, ou une intervention, à un résultat souhaité. Une règle technologique vérifiée est une règle qui s'est révélée effective, et ce de manière systématique, à l'intérieur du contexte de son utilisation projetée. Elle peut avoir un fondement scientifique.

Pour MARCH et SMITH, et HEVNER et al. (2004), la « science-conception » consiste en deux activités de base : construire et évaluer. Construire correspond à fabriquer un artéfact ou une innovation dans un but précis, alors qu'évaluer consiste à juger les performances de l'artéfact conçu. Evaluer signifie aussi bien développer des critères d'évaluation, que confronter l'artéfact à ces critères. Pour JARVINEN (2004) Le processus séquentiel couvre la phase de construction de l'objet, son utilisation et sa démolition éventuelle.

Le processus de construction permet de passer d'un état initial à un état final souhaité. Si l'état final n'est pas connu, il est possible de le définir dans un premier temps, puis d'implémenter des mesures pour l'atteindre, ou alors de réaliser en parallèle la recherche d'objectifs et l'implémentation. C'est à ce dernier type de méthode que nous avons eu recourt dans le cadre de notre travail de mémoire. Nous avions

préalablement établi les spécifications décrites dans le cahier des charges du logiciel, mais nous en avons aussi créées d'autres au fur et à mesure de notre avancée dans la conception. Ce procédé présente les avantages de permettre d'imaginer des choses qui n'existent pas encore. Nous savions par exemple que nous souhaitions développer un programme qui permette de mettre en lumière des réseaux sociaux à l'aide des données présentes sur les pages personnelles, et nous avions déjà formulé certaines de nos attentes quant à ses fonctionnalités. Certaines idées sont venues par la suite se greffer à ces spécifications de base, au fur et à mesure que les propriétés de ce que nous construisions se dessinaient. Nos connaissances techniques étaient sommaires au début de nos recherches, et nous envisagions le code PHP comme un moyen de conception et non une fin en soi. A plusieurs reprises, le comportement des algorithmes que nous avons développés ont dépassé nos prévisions, et sont venus alimenter de nouvelles réflexions.

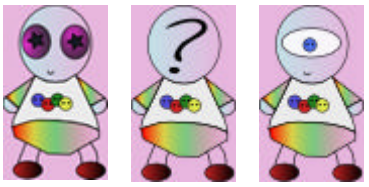
JARVINEN (2004) s'est attaché à définir les caractéristiques de la recherche « *science-conception* ». Nous allons tenter de démontrer en quoi notre démarche méthodologique a été sous bien des aspects similaire à cette acception. Pour JARVINEN (Ibid, 2004), la « *science-conception* » doit produire un artefact viable (construit, modèle, méthode ou instanciation), elle doit développer une solution technologique à des problèmes pertinents. Notre travail tente de répondre à un besoin réel, celui de trouver des informations sur les gens et leurs réseaux sociaux, des personnes partageant notre intérêt pour quelque chose, ou encore des ressources pour l'aide à apprentissage. La problématique touche à des aspects technologiques complexes, telle que la recherche d'information ou l'indexation de documents, mais comporte des aspects relatifs au domaine des sciences sociales et cognitives. La viabilité de l'artéfact final produit peut bien sûr être questionnée, néanmoins, tel était notre objectif de départ. Comme le note JARVINEN (Ibid, 2004), L'artefact produit doit être rigoureusement évalué en termes d'utilité, de qualité et d'efficacité. La science-conception doit fournir des contributions claires et vérifiables dans le domaine du design, et s'appuie sur l'application de méthodes rigoureuses de construction et d'évaluation. Nous nous sommes attachés jusqu'ici à décrire clairement le procédé de développement de notre programme, que nous tenterons par la suite d'évaluer. JARVINEN précise que la recherche d'un artefact efficace doit se servir des moyens disponibles pour atteindre des fins désirées en obéissant aux lois d'un environnement problématique. Là aussi, si certains aspects de notre programme sont inédits, nous avons fait usage de certaines portions de codes produites par d'autres, afin de faire une économie de temps, et parce que nous jugions que cette solution était la plus efficace. JARVINEN reprend les propositions de critères évaluation de prototypes de MARCH et SMITH (1995) dont la complétion, la simplicité, l'élégance, la compréhensibilité et la facilité d'utilisation, et rappelle l'importance de la communication et de la cognition (BOLAND et TENKASI, 1995). Dans un souci de rigueur scientifique, nous nous efforcerons de comparer notre prototype à ces différents critères de mesure dans le chapitre consacré à l'évaluation préliminaire. Cette évaluation systématique ne correspond cependant pas à notre priorité, puisque notre travail, de par certains de ses aspects inédits, ne peut pas toujours être comparé à des solutions préexistantes. Ainsi, pour MARCH et SMITH (1995), lorsque l'on cherche à créer un objet qui n'existe pas encore sous une autre forme, la « *contribution à la recherche consiste en la nouveauté de l'artefact et dans la capacité à persuader qu'il est efficace. L'évaluation des*

performances n'est pas nécessaire à ce stade. (…) Le chercheur peut interroger les bénéfices potentiels de l'objet nouvellement produit en soulignant ses aspects utiles. » (notre traduction).

5. Implémentation

Sur le plan architectural, notre programme consiste en 3 composants distincts accessibles à l'adresse : http://tecfa.unige.ch/perso/staf/genet/lsa/socnetminer.HTML

-le générateur de profils FOAF
-l'interrogateur de FOAF
-le visualiseur de FOAF



Ces différents composants sont accessibles de manière centralisée depuis une page Web.

Nous avons choisi PHP 5 pour développer les différentes applications, à la fois pour sa facilité d'utilisation et pour ses riches fonctionnalités. Rappelons que nous visions essentiellement l'implantation d'une innovation utile sur le plan sociale et pédagogique, et non l'élégance dans la programmation. Nos connaissances en programmation étant par ailleurs restreintes, les méthodes que nous avons utilisées ne sont certainement pas les plus simples, les moins coûteuses en termes de ressources, ni les plus élaborées sur le plan stylistique.

PHP nous a servi de médium afin de créer un prototype d'interface. Certaines fonctionnalités n'ont été implémentées que partiellement faute de temps. Dans ces cas-là, nous tentons de décrire précisément comment de quelle manière nous avions pensé les introduire. Nous nous attachons à décrire les choix d'implémentation de chaque sous-composant de notre système dans ce chapitre.

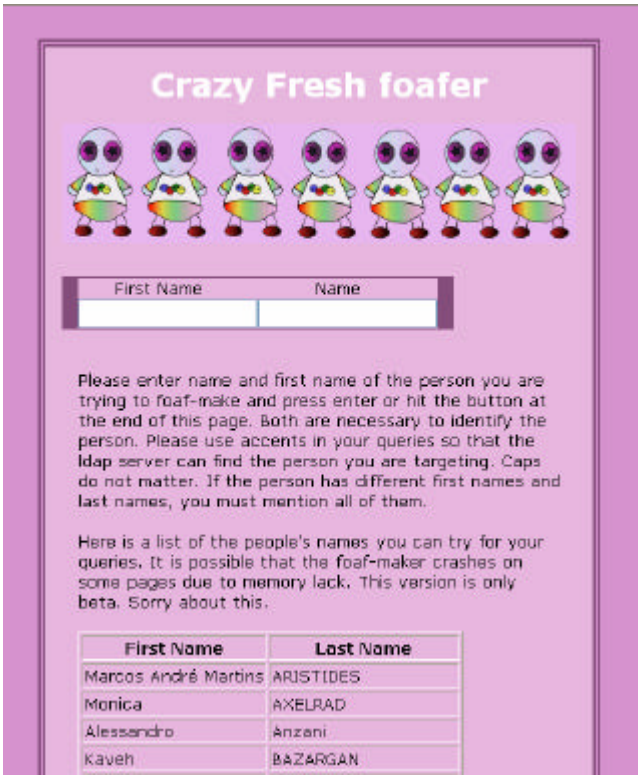
5.1 Le générateur de profils FOAF

Le générateur de profil FOAF était la pièce maîtresse de notre système, il devait réunir les différents types d'informations disponibles sur les pages personnelles de membres du TECFA, et les transformer en données structurées à l'aide du langage FOAF. Il devait être capable de dialoguer avec le serveur LDAP de l'université, d'extraire des données textuelles et des liens de pages personnelles aux formats HTML et XML, de préparer les données textuelles, de les analyser et de les indexer, de reconnaître les personnes mentionnées dans le texte, de convertir les résultats en langage FOAF et de les afficher sous une forme lisible par les humains.

Nous avons opté pour une interface simple : l'interface du générateur de FOAF (« foaf maker ») est composée de deux champs dans lesquels on peut indiquer le nom et le prénom de la personne dont on souhaite générer le profil. Il s'agit d'une page PHP qui s'appelle elle-même. Le nom et le prénom de la requête sont stockés dans des variables. La connexion et l'extraction des informations du serveur LDAP n'a pas représenté de grande difficulté. Nous nous sommes servi des fonctions LDAP intégrées dans PHP 5. Nous nous connectons de manière anonyme au serveur LDAP de l'université et recherchons les personnes du même nom et prénom que ceux envoyés par le biais du formulaire. Nous extrayons ensuite l'identifiant unique, l'adresse des pages personnelles et le groupe d'appartenance de la personne.

La préparation des données textuelles s'est révélée plus ardue que nous ne l'avions envisagée. En effet, le programme devait réunir toutes les pages produites par la personne, ce qui sous-tendait la capacité à différencier des pages appartenant au même dossier étudiant des pages du World Wide Web. Sur le serveur Web du TECFA, chaque personne possède un dossier personnel dans lequel elle peut mettre les documents qu'elle souhaite publier. Nous avions songé à utiliser les données situées dans ces dossiers, mais cela posait des problèmes de respect de la sphère privée. En effet, la plupart des étudiants ne sont pas conscients que les pages qui ne sont pas reliées à d'autres pages sont tout de même accessibles depuis Internet, et tendent à se servir des dossiers Web comme des répertoires personnels. C'est pourquoi nous avons préféré concevoir une fonction d'extraction des liens, qui fonctionne par récursions successives tout en éliminant les liens vers des pages externes et les pages déjà ouvertes. Le programme cherche l'adresse de la page personnelle qu'il a extraite du serveur LDAP, il en extrait le contenu. Il stocke tous les liens sur la page et différencie les liens vers d'autres pages du dossier étudiant, les liens vers l'extérieur et les liens vers des pages déjà parcourues. Pour ce faire il établit quelle est la racine du dossier étudiant à l'ouverture de la page personnelle principale et compare les autres pages à cette structure pour s'assurer qu'elles appartiennent bien au site de l'étudiant. Les liens sont repérés grâce à une expression

régulière. Le programme opère une récursion et ouvre toutes les pages qui appartiennent au dossier. Chaque page ouverte est stockée dans un tableau. Il extrait tous les liens de chacune des pages et opère une récursion si les pages n'ont pas encore été ouvertes. Chaque nouvelle page ouverte est comparée au tableau des pages déjà ouvertes pour évaluer si la récursion doit être faite ou pas. Le programme s'arrête quand il a crawlé toutes les données et renvoie un tableau contenant les adresses de toutes les pages locales au site de l'étudiant. Un autre écueil tenait en la présence de liens relatifs au sein des pages. Nous avons dû améliorer notre fonction de manière à lui faire assurer la conversion de tous les liens relatifs en liens absolus.



Une fois les sources d'information disponibles sur la personne circonscrites, le programme devait soumettre les données à une analyse lexicométrique, et si possible à un algorithme d'extraction de la sémantique latente. Il se base sur le tableau des liens locaux, ouvre chaque fichier pour en extraire le code HTML. Une fois le code extrait, les balises HTML sont nettoyées, et les accents et les symboles sont supprimés. Chaque mot est comparé à une liste de mots vides et les parasites sont éliminés par une recherche à l'aide d'une expression régulière. Le corpus ainsi constitué est écrit dans un fichier stocké dans un répertoire et nommé selon le nom de famille de la personne. Chaque mot est séparé par un retour de chariot. A chaque sollicitation du programme, celui-ci ouvre tous les fichiers corpus et les retranscrit les uns après les autres dans un fichier corpus général, séparés par un espace blanc. Ce fichier sera par la suite utilisé pour l'indexation, mais nous reviendrons sur ce point dans le paragraphe suivant. On compte finalement les mots utilisés par la personne et leurs occurrences. Le décompte des occurrences se fait grâce à une boucle : chaque mot est stocké dans un tableau, et chaque terme est comparé aux autre, à chaque fois qu'un doublon apparaît, le compteur d'occurrences est incrémenté. Tous les mots fréquents (utilisés plus de 3 fois) sont placés dans un tableau à part.

Il existe une balise « *foaf:topic_interest* » en langage FOAF qui permet de décrire les centres d'intérêts des gens. Cette balise était centrale à notre réflexion puisque c'est elle qui permet de résumer les domaines d'expertise et les préoccupations des gens. Un des points nodaux de notre générateur de FOAF tenait à l'extraction de données textuelles pertinentes pour remplir cette balise. Nous souhaitions utiliser un algorithme de sémantique latente dans le but d'extraire les mots les plus représentatifs du texte. Trouver un algorithme LSA utilisable nous a pris énormément de temps. Nous avons choisi un algorithme LSA de Bellcore technology, parce qu'il s'agit du plus complet que nous avons eu la possibilité de tester, et parce qu'il permet les comparaisons entre plusieurs documents distincts et non seulement entre des termes ou de termes à document. Le module LSA est installé sur le serveur de l'Université et répond à des requêtes en Shell Unix effectuées depuis PHP 5. L'indexation des documents se fait en comparaison du corpus constitué par l'ensemble des mots figurant sur les pages personnelles des personnes dont le profile a préalablement été généré (fichier corpus général mis à jour à chaque fois que le programme est activé). Le générateur de FOAF interroge le programme LSA pour connaître les mots au sein de la production de la personne qui obtiennent un score LSA les plus élevés en comparaison de l'ensemble du texte produit par la personne. Nous avions placé un seuil de significativité arbitraire. Nous avons finalement renoncé à une telle implémentation, parce que les résultats obtenus semblaient empiriquement moins pertinents que ceux obtenus par simple tri d'occurrences. Pour illustrer ce propos, lorsque nous interrogions la base de fichiers FOAF pour savoir qui était intéressé par l'éducation au TECFA (une sous-unité de la faculté des sciences de l'éducation !), nous n'obtenions aucun résultat positif. Avec un simple tri par occurrences limité arbitrairement à 3, nous identifiions plusieurs personnes concernées par le sujet. Nous avons donc abandonné l'implémentation de l'algorithme LSA dans le générateur de FOAF et avons préféré l'utiliser pour améliorer l'interrogateur de FOAF par des synonymes des mots recherchés et pour illustrer la proximité entre les corpus des personnes dans le visualiseur.

Une fois les données au format HTML exploitées, nous disposions d'une autre source d'informations. En effet, au TECFA, chaque étudiant et chaque membre du corps enseignant est encouragé à mettre en ligne ses travaux sur une page Web, qui sert de portfolio des productions personnelles de la personne. Les adresses de ces pages étant répertoriées par le serveur LDAP de l'université, leur accès ne présentait donc pas de grande difficulté. Néanmoins, la majorité des pages travaux des étudiants étaient publiées au format de données XML. Ce format de données structuré présentait l'avantage de nous éviter de générer automatiquement des métadonnées approximatives sur le contenu textuel, puisque les données étaient déjà insérées dans des méta-balises. La problématique tenait plutôt à l'extraction des données de ces balises. Les grammaires utilisées pour les balises XML (DTD) divergeaient aussi d'une promotion d'étudiant à l'autre, c'est pourquoi notre code devait s'adapter en fonction des différents cas de figures. Nous avons choisi d'extraire les données avec XPath, parce que cette méthode était de loin la plus simple, et que nos requêtes n'étaient pas suffisamment complexes pour exiger l'usage du DOM (Document Object Model). Le programme recherche les balises de login UNIX et pseudonyme Moo selon les formalismes qui correspondent aux différentes promotions (par exemple « *<moo-login>* », ou « *<moo>* »). Les pages travaux fournissent des liens vers les différentes productions des étudiants sous forme de données XML, c'est pourquoi le stockage des liens dans des balises « *foaf:made* » était aisé.

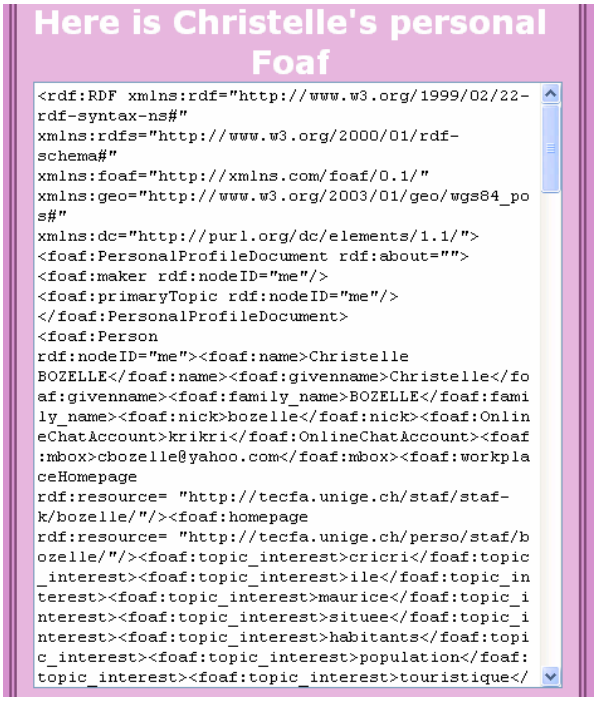
Ainsi, le programme prend tout ce qui se trouve dans les balises « exercice » et le stocke dans le tableau des productions personnelles de la personne.

Pour investiguer le réseau social de la personne, le programme recherche dans le texte extrait des pages HTML et XML les noms et prénoms de personnes qui sont stockés dans la banque de fichiers FOAF. Les fichiers FOAF contenus dans cette banque sont générés à chaque fois que le générateur de FOAF fonctionne. Nous expliquons plus loin le procédé de création des fichiers.

A chaque fois qu'il tourne, le programme met à jour la liste des fichiers contenus dans la banque (« scutter plan »). Les fichiers FOAF sont nommés selon le nom de famille de la personne. Dans un premier temps, le programme se base sur le scutter plan pour ouvrir tous les fichiers RDF les uns après les autres. Il place les noms des fichiers dans un tableau. Les balises « *foaf :givenname* » et « *foaf :family_name* » sont extraites de chaque fichier par requête Xpath et stockées dans deux tableaux Un tableau à trois dimensions répertorie les noms, prénoms et adresses des fichiers FOAF des personnes. Dans un deuxième temps, chacune des valeurs des tableaux est comparée au tableau des mots figurant sur la page personnelle ou la page travaux. Si un terme figure dans les deux tableaux, qu'il est différent du nom de la personne dont on génère le profil et qu'il apparaît pour la première fois, le nom de la personne, le prénom et le nom du fichier FOAF correspondant sont stockés dans le tableau des amis de la personne. Enfin, le programme ouvre chacun des fichiers stockés dans le tableau des amis et effectue une requête Xpath pour en extraire les balises « *foaf :name* » (nom foaf) et « *foaf :mbox* » (adresse e-mail).

```
- <rdf:RDF xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:foaf="http://xmlns.com/foaf/0.1/"
    xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  - <rdf:Description>
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/bozelle.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/genet.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/tassini.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/mudry.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/sylvain.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/gorga.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/boucheri.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/claude.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/court.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/decastro.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/diego.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/hocquet.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/kapusova.rdf" />
      <rdfs:seeAlso rdf:ressource="http://tecfa.unige.ch/perso/staf/genet/lsa/foaf/lavarini.rdf" />
```

En dernier lieu, le générateur de profils créée le fichier RDF. Le programme se sert des différentes variables extraites jusqu'alors pour remplir les balises FOAF *« foaf :name »*, *« foaf :givenname »*, *« foaf :family_name »*, *« foaf:nick »*, *« foaf:OnlineChatAccount »*, *« foaf:mbox »*, *« foaf:workplaceHomepage »*, *« foaf:homepage »*, *« foaf:group »*, *« foaf:topic_interest »*, *« foaf:made »*, *« foaf:knows »*. Il écrit l'en-tête du fichier, les déclarations d'espaces de noms, les balises et leurs contenus respectifs dans un fichier texte et affiche les données dans une balise HTML « *zone de texte* » (« *textarea* ») pour permettre l'affichage du code dans le navigateur. Un lien vers une visualisation du fichier FOAF est renvoyé (foaf Explorer), ainsi que vers le scutter des différents fichiers FOAF générés.



Dans un souci d'économie de ressources, nous avons ajouté une fonction de détection de l'âge des fichiers RDF. Le programme examine la dernière date de modification du fichier RDF et, si elle est inférieure à deux semaines, il renvoie un lien vers la visualisation de ce fichier sans exécuter les autres actions.

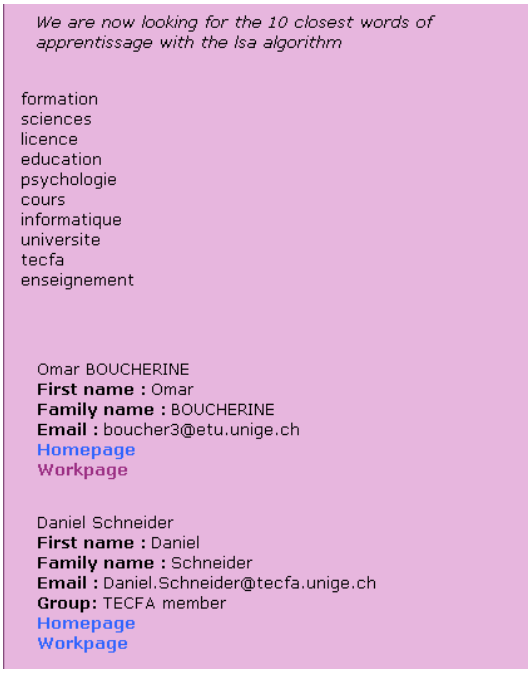
5.2 L'interrogateur de fichiers FOAF

L'interface de l'interrogateur de fichiers FOAF est constituée de deux sous-modules. Le premier sous-module permet d'interroger les centres d'intérêts des personnes dont les profils ont été générés à l'aide du générateur de fichiers FOAF (« *Who is interested in what ?* »). Le second permet de questionner les liens sociaux des personnes. Il s'agit de deux formulaires semblables, composés de trois champs dans lesquels on peut entrer le nom, le prénom ou le nom et le prénom de la personne. Les formulaires envoient les variables à un script PHP.

Le « *Who is interested in what* » (« *qui est intéressé par quelque chose* »), autrement dit l'interrogateur de centre d'intérêts, est divisé en deux parties. Il permet, soit de savoir qui partage un centre d'intérêt donné, soit quels sont les centres d'intérêts de quelqu'un. Deux cases à cocher peuvent être activées pour obtenir tous les détails concernant une personne (y compris son adresse e-mail et les adresses de ses pages personnelles ou son groupe d'appartenance). On peut entrer un centre d'intérêt et rechercher parmi les fichiers FOAF les personnes qui partagent probablement ce centre d'intérêt, ou entrer le nom, le prénom ou les deux de quelqu'un et obtenir la liste de ses centres d'intérêts.

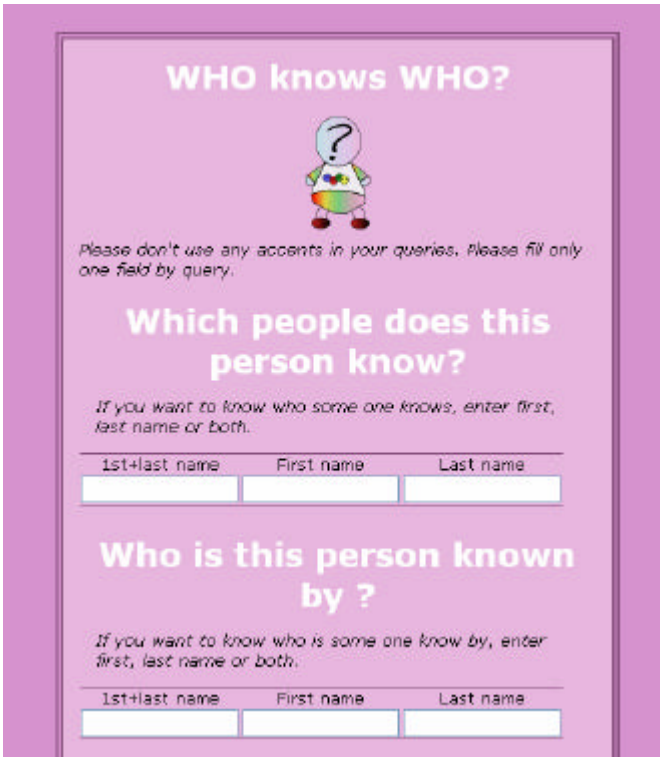
Le fichier PHP appelé par le formulaire examine les variables qui lui sont transmises. Il exécute des fonctions différentes en fonction des champs qui sont remplis ou des cases cochées. Si le champ des centres d'intérêts est rempli, le programme ouvre chacun des fichiers de la banque de fichiers FOAF et effectue une requête Xpath pour en extraire le nom et un tableau des centres d'intérêts. Le contenu du champ intérêt est comparé au tableau des centres d'intérêt, et si une concordance est détectée, le nom de la personne concernée est stocké dans un le tableau des gens qui partagent un intérêt. Si la case des informations détaillées est activée, alors le programme effectue aussi des requêtes Xpath pour extraire les balises d'adresse e-mail, de page personnelle, de page travaux et de groupe d'appartenance.

Si la case « synonymes LSA » est cochée, le programme interroge l'algorithme LSA installé sur le serveur de l'université par le biais de commandes de terminal UNIX, et renvoie les 10 mots les plus proches sur le plan sémantique. Il compare chacun de ces synonymes avec les centres d'intérêts des fichier FOAF, si il y a concordance avec un des tableaux, il extrait le nom et éventuellement les informations détaillées sur la personne.



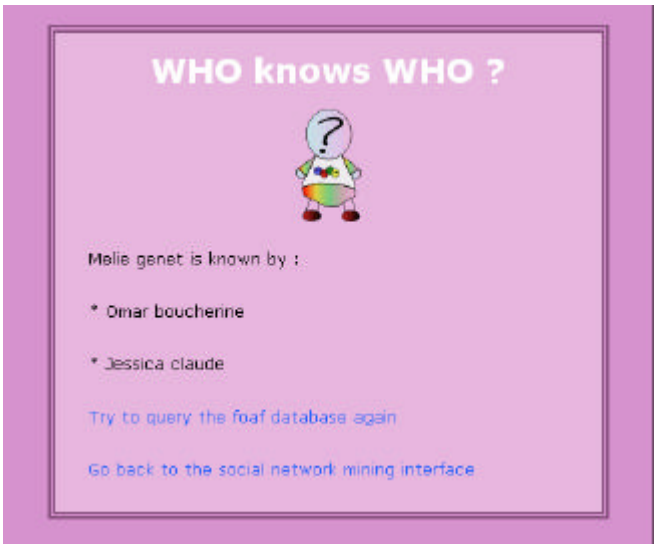
Si les champs du nom, prénom ou nom+prénom sont remplis, le programme ouvre les fichiers FOAF les uns après les autres et contrôle si les contenus des variables *« foaf :family_name », « foaf :givenname » et « foaf :name »* correspondent. Si c'est le cas, le programme renvoie le contenu des balises *« foaf :interest »* du fichier FOAF correspondant.

Le deuxième module de l'interrogateur de FOAF, le « *Who Knows Who* », permet prendre connaissance du réseau de relation des personnes. On peut interroger les fichiers FOAF pour savoir quelles sont les personnes que les gens connaissent, et si ces relations sont bidirectionnelles.



Si les champs noms, prénom, ou prénom+nom de la partie du haut sont remplis, le programme ouvre les fichiers FOAF de la banque de FOAF successivement et contrôle si les balises FOAF correspondantes sont identiques à la requête. Si c'est le cas il renvoie le contenu des balises *<foaf:knows> <foaf:Person> <foaf:name>* ».

Si les champs noms, prénom, ou prénom+nom de la partie du bas sont remplis, le programme ouvre les fichiers FOAF et extrait le contenu des balises « *foaf :name* » et « *<foaf :knows> <foaf :Person> <foaf :name>* ». Si le contenu des balises « foaf :knows » contient le nom de la requête, alors il place le foaf :name du fichier correspondant dans la liste des connaissances de la personne (« in-links » sociaux).

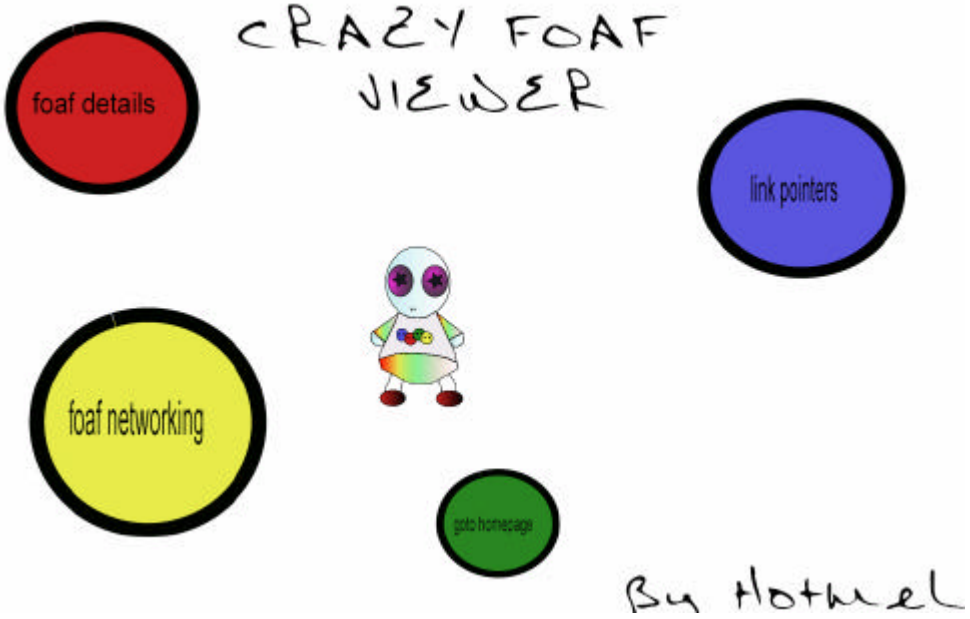


5.3 Le visualiseur de FOAF

Le dernier module du système, le « *foaf :viewer* » permet de visualiser les connaissances d'une personne. L'interface est similaire à celle du générateur de profils ou de l'interrogateur.

Un formulaire PHP reçoit les variables, ouvre les fichiers FOAF de la banque de FOAF et les compare aux contenus des balises « *foaf :givenname* », « *foaf :family_name* » et « *foaf :name* ». S'il y a des similitudes, alors le programme repère l'adresse du fichier correspondant.

Plusieurs solutions s'offraient à nous pour la génération de la visualisation des fichiers FOAF. Nous avons opté pour la génération de code SVG (Scalable Vector Graphics) par le biais de l'application d'une feuille de style XSLT (Extensible Stylesheet Language Transformation) aux fichiers RDF. Cette méthode nous a paru particulièrement adaptée pour la mise en forme des fichiers FOAF, le RDF étant une sous-classe de XML. L'implémentation du DOM XSLT nous permettait d'effectuer des requêtes complexes sur les fichiers FOAF et de manipuler les données que nous souhaitions mettre en avant sur le plan graphique.



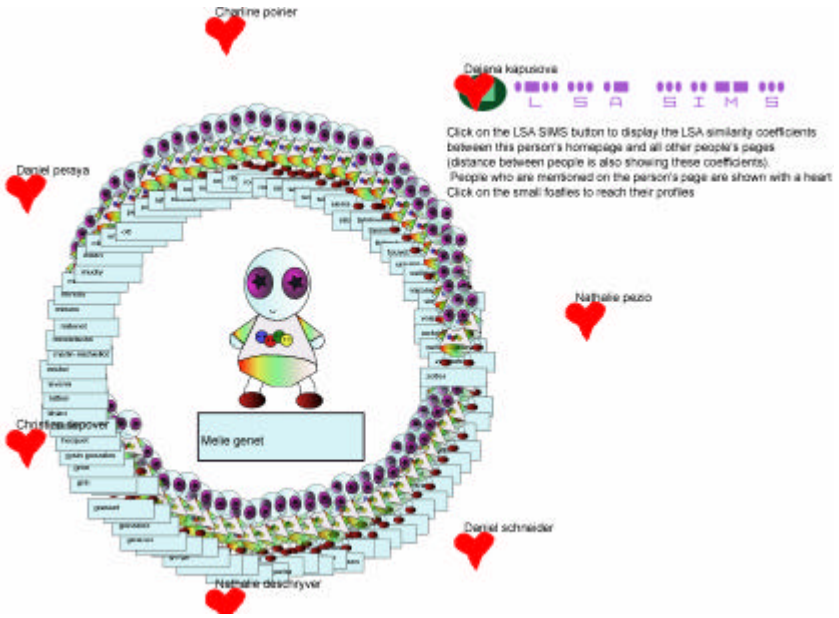
Ainsi, notre programme applique une feuille de style XSL au fichier FOAF de la personne dont le nom a été entré dans le champ de l'interface, grâce à une instanciation du processeur DOM XSLT de PHP 5. L'output se fait sous forme d'un graphique SVG. La feuille de style XSL génère une grande figurine pour représenter la personne.

Une interface en SVG dynamique permet d'accéder à différents sous-menus qui affichent les détails du fichier FOAF de la personne, une visualisation de l'ensemble des personnes du TECFA et des degrés de proximité de leurs productions verbales (comparaison de la similarité LSA) avec la personne, un lien vers sa page personnelle et une visualisation des liens qui pointent vers l'extérieur de sa page. Nous n'avons pas développé cette dernière fonctionnalité faute de temps.

Lorsque l'on clique sur le bouton « *Foaf details* », le programme affiche le contenu des balises FOAF les plus courantes.

Le bouton « *goto homepage* » renvoie directement à la page Web du sujet.

Lorsque l'on clique sur « *Foaf-network* », le programme va effectuer des requêtes XPath sur un fichier généré automatiquement à l'entrée des données dans le formulaire du foaf-viewer. A chaque fois qu'un nom est envoyé par le formulaire, le script PHP fait appel à l'algorithme LSA pour comparer le corpus textuel de cette personne avec celui de chaque autre personne du TECFA. Il ouvre tous les fichiers FOAF pour en extraire les noms et génère un fichier XML dans lesquels sont mises en relation les personnes et le coefficient de similarités de leur corpus avec celui du sujet. Le programme affiche une grande figurine au centre et des petites pour représenter les autres personnes. Le logiciel recherche les balises *foaf :knows* dans le fichier FOAF du sujet et génère un cœur pour chaque personne connue. On peut accéder au foaf-viewer de chaque autre personne du TECFA en cliquant sur les petits bonshommes. Un bouton « LSA similarity » permet d'afficher les coefficients sous forme numéraire.



Nous souhaitions développer un module d'affichage des profils FOAF avec un output en XHTML. Nous avions pensé faire fonctionner le générateur de FOAF avec une base de données mySQL et stocker de cette manière les « oulinks » (liens externes) des personnes. Nous aurions alors pu questionner la base pour générer un autre type de visualisation mettant en évidence les liens d'une personne avec le reste du Web. Ces fonctionnalités n'ont pu être développées faute de temps, et ne figurent en l'état actuel que sous forme d'une icône inactive dans le SVG dynamique du « foaf :viewer ». La génération du SVG se basant sur une centaine de fichiers, le dispositif met énormément de temps à répondre actuellement. La visibilité des différents protagonistes est mauvaise et exigera un travail de développement ultérieur. Ces aspects négatifs feront l'objet de nos prochains efforts.

6. Evaluation préliminaire

Nous nous restreindrons à une évaluation succincte du prototype implémenté, dans la mesure où son ergonomie et son usabilité n'ont pas été approfondies par manque de temps. Nous souhaitions avant tout mettre en place un projet de système innovant pour automatiser la recherche de pairs dans un contexte d'apprentissage, qui mette exergue les caractéristiques inter- et intra-personnelles des membres de l'université. Nous tâcherons de démontrer les aspects d'utilité de ce système et ses applications possibles dans le chapitre qui suit.

Reprenons chacun des critères d'évaluation des prototypes proposés par MARCH et SMITH (op. cit. 1995) et observons notre système sous leur jour. La complétion ne correspond pas à l'un de ses points forts, puisque nous ne disposions de quelques mois pour élaborer une proposition de programme. Certaines fonctionnalités sont restées des projets d'implémentation. En particulier, nous aurions vivement souhaité interfacer le foaf-maker avec une base de données mySQL pour contourner certains échecs dus à un épuisement de la mémoire vive, et finaliser le foaf-viewer.

En termes de simplicité, le système est défendable parce qu'il repose sur un langage léger facile à traiter : le RDF, interrogé à l'aide de requêtes SimpleXML/Xpath. D'autres aspects sont cependant plus alambiqués, comme le recourt à l'algorithme d'analyse de la sémantique latente ou la méthode de génération du SVG dynamique à l'aide du DOM XSLT. Nous nous sommes efforcés de conserver un design relativement sobre pour la conception des interfaces des différents modules, dans l'espoir de faciliter leur utilisation.

Seul un passage au banc d'essai d'utilisateurs pouvait nous fournir des indications objectives quant au degré de compréhensibilité et à la facilité d'utilisation du logiciel. Afin de mieux prendre conscience de ces aspects, nous avons soumis l'interface à un panel de 5 utilisateurs. Nous leur avons donné l'adresse du logiciel et leur avons demandé d'accomplir des tâches avec le programme. Ils devaient trouver quels étaient les centres d'intérêts de certaines personnes de l'université, quelles personnes elles connaissaient et qui les connaissaient. Nous n'avons pas soumis le visualiseur de FOAF au banc d'essai, le travail de

développement du module n'étant pas suffisamment finalisé. Les remarques de utilisateurs ont toutes été positives et ils sont parvenus au bout de toutes les activités. Cependant, ce test a permis de mettre en lumière certains problèmes d'utilisation des interfaces. Par exemple, les accents et les types d'encodage donnent lieu à des confusions et des difficultés pour interroger les différents modules. En effet, lorsque l'on ne précise pas le type d'encodage dans un fichier XML (ou RDF comme ici), le codage utilisé par défaut est l'UTF-8. Or, UTF-8 ne supporte pas les accents français. Pour éviter les problèmes liés à l'encodage, nous avons supprimé les accents de toutes les données extraites des pages personnelles, ainsi que des données stockées dans les balises des fichiers FOAF. Le serveur LDAP de l'université fonctionne au contraire avec un encodage ISO, et conserve les accents. C'est pourquoi, dans l'interface du générateur de FOAF, il est nécessaire d'entrer les requêtes avec les accents, alors que dans les autres formulaires de l'interrogateur de FOAF, il faut au contraire les supprimer. L'effet d'habitude a notablement perturbé les utilisateurs, qui ont reproduit les mêmes comportements avec toutes les applications. Nous n'avons pas apporté d'effort particulier pour rendre l'utilisation des formulaires plus intuitive faute de temps. Il aurait été judicieux de prévoir un outil qui supporte les requêtes phonétiques ou partielles.

Un autre aspect améliorable soulevé par les utilisateurs tient à la taxonomie des champs proposés dans les formulaires. On propose d'entrer soit le nom (le nom dans le champ nom), soit le prénom (le prénom dans le champ prénom), soit les deux (le prénom et le nom de la personne dans le champ 1st name+last name). Cette distinction a une double fonction, tout d'abord les requêtes XPath sous-jacentes ne sont pas les mêmes : le champ nom permet de rechercher les balises « *foaf:family_name* », le champ prénom permet de rechercher les balises « *foaf:givenname* » alors que le champ 1st+last name interroge les balises « *foaf:name* ». Elle est donc impliquée par la structure du langage FOAF. Deuxièmement, elle permet d'identifier de manière exclusive quelqu'un qui posséderait des homonymes de nom ou de prénom. Plutôt que de créer un troisième champ prénom+nom, nous aurions dû autoriser les requêtes associant des champs multiples. Après avoir soumis notre travail à cette évaluation préliminaire, nous allons maintenant tenter de mesurer ses perspectives et ses limites.

7. Discussion

Dans le cadre de notre travail de mémoire de DESS, nous avons cherché à développer des outils qui permettent de résumer les informations contenues dans les pages personnelles des membres de l'université, et qui facilitent la mise en évidence de réseaux sociaux et de communautés de pratiques à partir de ces données. Nous avons implémenté une interface composée de trois sous-parties. Le premier module permet de générer automatiquement des profils composés de métadonnées lisibles par les machines à partir des données contenues dans les pages personnelles des membres de l'université. Le deuxième module permet de rechercher des informations sur les gens contenues dans la banque des profils générés par le premier module. Le dernier module propose un mode de visualisation de certaines informations contenues dans les profils, dont les personnes qu'ils connaissent.

Ces outils ne sont que des prototypes à l'heure actuelle, mais nous pensons qu'ils pourraient servir de support efficace à la localisation des connaissances et des communautés d'intérêts ou de pratiques. Nous avions pour ambition première d'appréhender des caractéristiques distinctives ou des communautés de personnes grâce aux données publiées sur des pages personnelles. Nous pensons avoir atteint cet objectif, dans la mesure où les outils développés dans le cadre de travail permettent d'obtenir des indices sur les préoccupations des membres de l'université et leurs réseaux de connaissances. L'évaluation préliminaire du programme a montré que des utilisateurs auxquels nous avons soumis le logiciel ont pu trouver relativement aisément des informations sur nos centres d'intérêts et nos amis. Le système est cependant critiquable sous certains aspects. Nous faisons état des limites les plus évidentes ci-après.

Tout d'abord, le générateur de profils n'est fonctionnel que dans un environnement comme celui de l'université. Il ne peut être opérationnel que dans les cas où un serveur LDAP stocke des informations sur les gens, dont leur nom, prénom, e-mail, et surtout l'adresse de leurs pages personnelles. Nos connaissances concernant les technologies de détection de pattern dans les données (IR, Information retrieval, ou recherche d'information) étaient restreintes. Notre travail ne prétend pas atteindre le niveau d'un travail de spécialiste en intelligence artificielle. Nous aurions pu nous concentrer sur l'élaboration d'algorithmes probabilistes de détection des noms et prénoms des personnes, mais nous avons préféré faire usage des données disponibles sous forme déjà structurée. Nous n'avons pas utilisé de méthodologie de développement orientée objet, ni visé l'interopérabilité ou l'extensibilité applicative. Nous pensons néanmoins que certaines des méthodes utilisées au sein de l'implémentation sont porteuses pour l'avenir, plus particulièrement l'association des technologies FOAF/SimpleXML et FOAF/DOM XSLT.

En outre, le générateur de profil est basé sur une méthode de création de métadonnées relativement grossière. A ce sujet, nous nous sommes interrogés quant à l'implication de la génération de métadonnées de mauvaise qualité. Premièrement, n'était-ce pas aller à l'encontre des efforts de la communauté du Web sémantique pour améliorer le Web en élaborant des métadonnées précises sur les contenus ? Si l'humain peut communiquer des informations sémantiques à la machine, l'inverse est un postulat assurément osé. Evidemment, le problème se pose aussi lorsqu'un humain tente de communiquer des informations sémantiques à un autre humain. Il existe toujours des variations entre les sens que l'un veut exprimer, et celui que l'autre va interpréter. Nous sommes conscients que la validité des métadonnées générées par notre système peut, et doit, être remise en question. Cependant, que notre méthode de génération de métadonnées n'est pas moins fiable que toute autre technique d'indexation. Chaque algorithme possède ainsi des limites et des avantages. Effectuer des requêtes dans plusieurs moteurs de recherche conduit en général à de meilleurs résultats, parce que leurs caractéristiques divergent. Très souvent, il arrive que les résultats fournis ne soient pas pertinents.

Un autre problème se pose, celui de la dualité des sources de représentation des gens. En effet, sur le serveur de l'université, nous avons d'une part créé notre profil FOAF manuellement, et d'autre part généré notre profil automatiquement grâce au FOAF maker. Deux profils existent donc de manière concomitante, présentant des informations différentes. Leigh Dodds, qui est à l'origine du premier

générateur de fichiers FOAF, le « *FOAF-O-MATIC* » utilise dans ses profils des balises RDF « admin » qui permettent de désigner le programme utilisé *(« <admin:generatorAgent RDF:resource="…"/> »)* pour générer le fichier FOAF, et la personne à contacter en cas d'erreur (« *<admin:errorReportsTo* RDF:resource="mailto :…"/> »). Il faut indiquer l'espace de noms « xmlns:admin=http://webns.net/mvcb/ » dans la déclaration d'espaces de noms. Si nous n'avons pas implémenté cette balise dans la version actuelle du programme, nous pensé qu'elle constitue une solution aux problèmes éthiques découlant de la génération automatique de métadonnées approximatives.

L'extraction des données à placer dans les balises « *foaf:topic_interest* » s'est révélée ardue parce que les deux méthodes que nous avons testées à cet effet souffrent des mêmes travers. En effet, tant la technique d'extraction des mots fréquents et l'algorithme d'analyse de la sémantique latente partent toutes deux du présupposé qu'un mot utilisé à plusieurs reprise est un mot important. Ce postulat, certainement vérifiable dans les cas de productions textuelles au format papier, ne l'est pas toujours dans le cas des pages Web. De nombreux termes sont ainsi répétés parce qu'ils font partie de la structure de navigation de la page Web, ou encore comme données codifiées (« dernière mise à jour de la page », « page conforme aux standards »). Nous avions tenté d'éliminer une partie de ces « mots vides » en notant les plus connus dans les termes de la « stoplist ». Il n'est bien sûr pas possible d'éliminer tous les mots appartenant à la structure des pages de cette manière, et les résultats s'en trouvent souvent biaisés.

Relativement à la recherche des personnes mentionnées dans les pages personnelles, le programme est incapable de faire la différence entre les personnes qui ont un nom de famille ou un prénom qui est aussi un nom commun, et le nom commun correspondant. Par exemple, il indiquera toujours qu'une personne qui dit qu'elle « *aime tailler la pierre à ses heures perdues* » connaît « *Pierre* ». Et de surcroît, il considèrera toujours que le « *Pierre* » dont il est question est le premier « *Pierre* » qui apparaît dans la liste des membres possédant un profil FOAF, et renverra les informations correspondantes. Lors du design de la fonction de recherche d'amis, nous nous trouvions devant le dilemme suivant : soit le programme trouvait une occurrence de « Pierre » et concluait que la personne connaissait tous les « *Pierre* », soit il s'arrêtait au premier « *Pierre* » de la liste. Dans les deux cas, le programme génère des données inexactes, mais nous préférions qu'il se trompe sur la personne mentionnée plutôt que sur le nombre de personne mentionnées dans une page. Nous reconnaissons qu'il s'agit d'un choix fait arbitrairement. En conséquence les personnes qui possèdent un nom de famille possédant une signification commune, et ceux qui possèdent des homonymes du même nom et figurent en premier dans l'annuaire se retrouvent en haut des « *charts* » sociaux. Il s'agit d'un autre biais introduit par le mode de génération des métadonnées. En outre, le générateur de foaf nomme les fichiers en fonction des noms de famille, ce qui empêche la prise en compte des profils de personnes qui portent le même nom de famille. Le plus récent écrase toujours le précédent et deux homonymes de noms ne peuvent coexister selon le système actuel. Cet aspect pourtant rectifiable ne l'a pas été faute de temps.

A ces considérations de fond s'ajoutent celles de forme, le système, tel qu'il est conçu actuellement, est très loin d'être pensé de manière optimale en termes d'économie de ressources. L'algorithme du

générateur de profils consomme beaucoup de mémoire vive et échoue dans certains cas où les pages qu'il parcourt comportent trop de vocabulaire à traiter, ou des liens trop nombreux. Cela est en grande partie dû à l'usage récurrent de tableaux. Nous souhaitons améliorer cet aspect dans les versions futures du logiciel. Une version couplée avec une base de donnée mySQL est d'ores et déjà en cours d'élaboration.

8. Conclusions

Nous cherchions à faire la proposition d'un outil de support à la perception de communautés d'intérêts et de pratiques basé sur la génération automatisée et l'interrogation de profils personnels à partir d'informations publiées dans les pages personnelles membres de l'université. Nous avons élaboré un prototype d'application permettant de donner des indications sur les centres d'intérêts et les réseaux sociaux des personnes. Cette interface, bien qu'elle ne soit pas pleinement finalisée, comporte cependant des aspects d'innovation technologiques intéressants, et pourrait représenter un outil de support pédagogique efficace. Des recherches ultérieures dans le domaine pourraient s'attacher à mesurer les effets de l'utilisation d'un tel outil sur une population d'apprenants, dans un contexte d'apprentissage collaboratif, ou encore approfondir les possibilités d'utilisation de l'analyse de sémantique latente pour montrer les proximités ou les divergences entre les productions textuelles des gens, et en faire émerger des communautés de pairs. Notre générateur de fichiers FOAF pourrait être aussi être modifié afin de parcourir des fils RSS ou d'autres types profils structurés utilisés communément dans les environnements de e-learning ou les portails.

9. Bibliographie et Webographie

ADAMIC, L., (1999). The small world Web, *Proceedings of the European Conf. on Digital Libraries*.

ADAMIC, L., EYTAN A., *Frequency of friendship predictors*,
http://www.parc.xerox.com/iea/papers/web10/

ALBA, R. (1972). SOCK. *Behavioral Science* 17.

DUQUENNE, V. (1993). GLAD. Paris: C.N.R.S.

ALBERT R., JEONG H., BARABASI, A.-L. (1999). The diameter of the World Wide Web, *Nature* 401.

ALLEN, C. 2004, October 13. Tracing the Evolution of Social Software._*Life with Alacrity*_. http://www.lifewithalacrity.com/2004/10/tracing_the_evo.HTML

ARNSETH, H.-C., LUDVIGSEN, S., WASSON, B. MORCH, A. (2001). Collaboration and Problem Solving in Distributed Collaborative Learning. *Proceedings of EuroCSCL. University of Maastricht*, Maastricht, the Netherlands 2001, pp 75-82.

AVIV, R., ERLICH, Z., RAVID, G., GEVA, A. (2003). Network Analysis of Knowledge Construction in Asynchronous Learning Networks. *Journal of Asynchronous Networks* 2003. 7: 1-23.

AVIV, R., ERLICH, Z., RAVID, G. (2004). Design and Mechanisms of Knowledge Constructing Online Learning Communities. *The 2nd Annual International MIT LINC Symposium &Workshop*, Cambridge, MA.

BARNES, J. (1954). Class and Committees in a Norwegian Island Parish. *Human Relations*, 7, 39-58.

BESTGEN, Y. (2004). Analyse sémantique latente et segmentation automatique des textes. *JADT 2004 : 7es Journées internationales d'Analyse statistique des Données Textuelles.*

BONIFACIO, M., CRUEL, R., MAMELI, G., NORI, M. (2002). A peer-to-peer architecture for distributed knowledge management, *Proceedings of the 3rd Symposium on Multi-agent Systems*. Large Complex Systems, and E-Businesses (MALCEB'2002).

BOURDIEU, P. (1986). *The forms of capital*.

BIENENSTOCK, E. BONACICH, P. (1997). Network Exchange as a Cooperative Game. *Rationality and Society* 1997. 9: 37-65.

BLAU, P. M. (1964). *Exchange and Power in Social Life*. Wiley, New York, NY.

BOCK, R. D., HUSAIN, S. Z. (1952). Factors of the tele: a preliminary report. *Sociometry.* 15, 206-219.

BOLAND, R. J., TENKASI R.V. (1995). Perspective making and perspective taking in communities of knowing, *Organization Science* 6, N°4.

BUNGE M. (1967*). Scientific Research I. The search for system.* Springer-Verlag, Berlin.

CHO, H., STEFANONE, M. GAY, G. (2002). Social Network Analysis of Information Sharing Networks in a CSCL Community. In Stahl, G. (Ed.) *Proceedings of Computer Supportfor Collaborative Learning (CSCL) 2002 Conference*. Lawrence Erlbaum, Mahwah, NJ, pp 43-50.

COATES, Tom. 2005, January 5. An addendum to a definition of Social Software. *_Plasticbag.org_*. www.plasticbag.org/archives/2005/01/an_addendum_to_a_definition_of_social_software.sHTML

DEERWESTER, S., DUMAIS, S. T., FURNAS, G. W., LANDAUER, T. K., HARSHMAN, R. (1990). Indexing by Latent Semantic Analysis. *Journal of the American Society For Information Science*, 41.

DIAZ, S. (2000). Cell Phone Signals Touted to Fight Traffic Wars, *San Jose Mercury News*, Jan. 20, http://www0.mercurycenter.com/svtech/news/indepth/docs/traf012100.htm

DOBSON, M., WENTING, M., DANIEL, H., CHAD, C. (2004). The role of social network representations in learning: TEAMVIEW as a reflective tool in learning environment. *Information Technology & Interactive Arts Program*, Simon Fraser University Surrey

EDWARDS, C. (2002). Discourses on Collaborative Networked Learning. *Networked Learning Conference*, Sheffield, UK 2002.

FLAKE G., LAWRENCE F., LEE GILES C. (2000). Efficient identification of web communities. In *Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Boston, MA, August 20-23, pp.150-160.

FREEMAN, L. C. (2000). Visualizing Social Networks, Carnegie Mellon: *Journal of Social Structure*.

Foaf-a-matic website : http://www.ldodds.com/foaf/foaf-a-matic.fr.HTML

GIBSON, D., KLEINBERG, J., RAGHAVAN P. (1998). Inferring Web communities from link topology, *Proceedings 9th ACM Conference on Hypertext and Hypermedia*.

GARTON, L., HAYTHORNTHWAITE, C., WELLMAN, B. (1997). Studying Online Social Networks. *JCMC* 3(1) http://www.ascusc.org/jcmc/vol3/issue1/garton.HTML

HAKKINEN, P., JARVELA, S., BYMAN, A., (2001). Sharing and Making Perspectives in Webbased Conferencing. In Dillenbourg, P., Eurelings, A. and Hakkarainen, K. (Eds.) *Proceedings of the First European Conference on Computer-Supported Collaborative Learning. Universiteit Maastricht*, Maastricht 2001, pp 285-292.

HALLORAN, J., ROGERS, Y. SCAIFE, M. (2002). Taking the 'No' out of Lotus Notes: Activity Theory, Groupware, and Student Groupwork. In Stahl, G. (Ed.) *Computer Support for Collaborative Learning: Foundations for a CSCL Community*. Lawrence Erlbaum, Boulder, CO 2002, pp 169-178.

HEVNER A. R., MARCH S. T., PARK, J. et RAM, S. (2004). Design science in information systems research, *MIS Quarterly* 28, No 1.

HOMANS, G. C. (1958). Social Behavior as Exchange. *American Journal of Sociology* 1958. 19: 22-24.

HomePageSearch : Search for Personal Home Pages of Computer Scientists http://hpsearch.uni-trier.de/hp/

InXight ThingFinder product page, http://www.inxight.com/products_wb/tf_server/index.HTML.

JARVINEN, P., (2004). *On Research Methods.* Juvenes Print, Tampere, Finland.

KAPLAN-LEISERSON, E. (2003). *We Learning : Social software and E-Learning* http://www.learningcircuits.org/2003/dec2003/kaplan.htm

KRACKHARDT, D., BLYTHE, J., McGRATH, C. (1995). *KrackPlot 3.0 User's Manual*. Pittsburgh: Carnegie-Mellon University.

LAKKALA, M., ILOMAKI, L., LALLIMO, J. et K., H. (2002). Virtual Communication in Middle School Students' and Teachers' Inquiry. In *Stahl, G. (Ed.) Computer Support for Collaborative Learning: Foundations for a CSCL Community.* Lawrence Erlbaum, Boulder, CO 2002.

LANDAUER, T. K., LAHAM, D. (1998) *Learning Human-like Knowledge by Singular Value Decomposition : A Progress Report.* Boulder, CO 1998.

LARSON, R. R., (1996). Bibliometrics of the World Wide Web: an exploratory analysis of the intellectual structure of cyberspace, Global Complexity: Information, Chaos and Control. T*he 1996 Annual Meeting of the American Society for Information Science*, October 21-26, 1996, Baltimore,Maryland, USA.

LAVE, J. WENGER, E. (1991). *Situated Learning : Legitimate Peripheral Participation*, Cambridge University Press.

LIN, N. (2001) *Building a Network Theory of Social Capital* (http://www.insna.org/Connections-Web/Volume 22-1/V22(1)-28-51.pdf)

LUNDBERG, G. A., STEELE, M. (1938). Social attraction-patterns in a village. *Sociometry*. 1, 375-419.

MARCH S.T., SMITH, G. F. (1995). Design and natural science research on information technology, *Decision Support Systems* 15.

MIKA, P., (2002). *Bootstrapping the FOAF-Web: An Experiment in Social Network Mining*.

MILGRAM, S. (1967).The small world problem, *Psychology Today* 1, 61.

MONGE, P. R. CONTRACTOR, N. S. (2001). Emergence of Communication Networks. InJablin, F. M. and Putnam, L. L. (Eds.) *New Handbook of Organizational Communication*. Sage, Newbury Park, CA 2001, pp 440-502.

MORENO, J. L. (1932). *Application of the Group Method to Classification.* New York: National Committee on Prisons and Prison Labor.

MORENO, J. L. (1953). *Who Shall Survive?* Beacon, N.Y.: Beacon House Inc.

NORTHWAY, M. L. (1952). *A Primer of Sociometry*. Toronto: University of Toronto Press.

PUTNAM, R. D. (1995) Bowling Alone: America's Declining Social Capital .*Journal of Democracy* (http://www.journalofdemo cracy.org/), January 1995.Volume 6, Number 1.

REFFAY, C., CHANIER, T. (2002). Social Network Analysis Used for Modeling Collaboration in Distance Learning Groups. In Cerri, S. A., Guarderes, G. and Paraguaco, F. (Eds.) *Lecture Notes in Computer Science (LNCS)* 2002, pp 31-40.

PROCTOR, C. (1953). Informal social systems. In C. P. Loomis, J. O. Moralis, R. A. Clifford, and O. E. Leonard (Eds.) *Turrialba*.(pp. 73-88). Glencoe, IL: Free Press.

ROGERS, E. M. (1986). *Communication Technology: The New Media in Society*. New York: Free Press.

SHARKES J., LANGHEINRICH, M., ETZIONI, O., (1997). Dynamic Reference Sifting: a Case Study in the Homepage Domain, *Proceedings of the Sixth International World Wide Web Conference*, pp.189-200

SHIRKY, Clay. 2004, October 6. Blog Explosion and Insider's Club: Brothers in cluelessness. *_Many-to-Many_*.
www.corante.com/many/archives/2004/10/06/blog_explosion_and_insiders_club_brothers_in_cluelessness.php

SOLLER, A., GUIZZARDI, R., MOLANI, A., PERINI, A. (2004). SCALE : Supporting Community Awareness, Learning, and Evolvment in an Organisationnal Learning Environment. *Proceedings of the 6th International Conference of the Learning Sciences,* Santa Monica, CA, 2004. International Society of the Learning Sciences (ISLS).

TCHERVSKY, A., KAHNEMAN, D. (1983). Extensional versus intuitive reasoning : The conjunction fallacy in probability judgment. *Psychological Review*, 90.

VAN AKEN, J. E. (2004). Management research based on the paradigm of the design sciences : The quest for the field-tested and grounded technological rules, *Journal of Management Studies* 41, No 2.

WANG, Y. FESENMAIER, D. R. (2003). Understanding the Motivation of Contribution in Online Communities: An Empirical Investigation of an Online Travel Community. *Electronic Markets* 2003. 13: 33-45.

WATTS D. STROGATZ S. (1998). Collective dynamics of smallworld networks, *Nature* 393, 440

WENGER, E. (1998). Communities of Practice : Learning, *Meaning and Identity*, Cambridge University Press.