

Les systèmes experts et la recherche en sciences humaines et sociales.

Cet article inaugure une série de tests approfondis effectués sur plusieurs dizaines (ou centaines) d'heures d'utilisation quasi-continue. Comme une automobile, un logiciel peut faire illusion quelque temps. Ensuite, on déchanté (rappelez-vous des articles sur Writer Plus lors de sa création...). C'est pourquoi, il a semblé à Icônes que des tests approfondis pouvaient rendre service à nos lecteurs qui comptent eux-aussi investir (!) de nombreuses heures de travail (ou de plaisir). Avec cette rubrique, ils sauront à quoi s'en tenir.

Turbo Expert : pour les experts pressés et fauchés...

Ne cherchez pas de beaux affichages : il n'y en a pas. Ne cherchez pas une gestion évoluée de l'ascenseur : on dirait un écran MS-DOS. Ne cherchez même pas à faire des "copier-coller" : ces fonctions sont inactives dans la plupart des cas.

Et pourtant, c'est un extraordinaire logiciel, à un prix imbattable.

Les systèmes experts (SE) pour le Mac sont nombreux. Depuis MacExpert, l'ancêtre (son créateur, Mindsoft, offre aujourd'hui des produits plus puissants) jusqu'aux programmes d'ACT-informatique en passant par ceux de Joy ou de G+M, pour ne citer que les sociétés françaises, nous aurons matière à de nombreux tests. Mais si nous commençons par Turbo-Expert, c'est surtout en raison de son exceptionnel rapport qualité-prix : **450 F** au lieu de 18 000 à 30 000 F pour ses concurrents.

Avant de citer les avantages (nombreux) et les inconvénients (rares) de Turbo-Expert, nous commencerons par une petite révision sur les systèmes experts.

Y a-t-il un expert dans la salle ?

Un système expert doit permettre à des « non-experts » de parvenir, avec l'aide de l'ordinateur, aux mêmes déductions que celles auxquelles l'expert humain serait parvenu.

Le principe de base apparaît simple : l'expert définit des règles du genre si une (ou plusieurs) condition est remplie, alors telle conséquence s'ensuit. A priori, il suffit donc de bien formaliser ses connaissances en un ensemble de règles et de sous-règles imbriquées les unes avec les autres. Le logiciel se charge ensuite de "chaîner" (en marche avant ou arrière) jusqu'à parvenir à la dernière d'entre elle qui est en principe la conclusion (dans le cas du chaînage avant). A ceux qui penseraient qu'une programmation en Basic pourrait suffire ("If... Then go to..."), nous ferons remarquer que les tests en cascade seraient extrêmement difficiles à gérer à partir d'un certain degré de complexité et d'enchevêtrement des conditions, d'où l'obligation de programmer des applications spécifiques prenant les règles comme des variables plus ou moins déclarées. Un autre argument faisant repousser une programmation classique tient justement à la non linéarité des raisonnements pouvant être simulés par un système expert (d'où la dénomination d'intelligence artificielle). Un système expert rend compte d'une connaissance qui n'est pas nécessairement organisée de manière séquentielle et totalement hiérarchisée (contrairement à un empilement de syllogismes). L'intérêt du traitement informatique tient à la possibilité de définir toute conséquence comme une nouvelle prémisse d'une nouvelle chaîne logique, ce qui permet de simuler (en partie) le raisonnement humain.

Le fonctionnement d'un système expert

Lorsqu'il est en mode "déduction", c'est-à-dire susceptible de remplacer l'expert vis-à-vis d'un utilisateur, le système va chercher toutes les prémisses et pose les questions correspondantes (avec des problèmes de syntaxe, cf. figure 1) et selon un mode binaire (réponses par oui/non).

A l'issue de cet interrogatoire de l'utilisateur, le logiciel examine tous les cheminements possibles pour tenter d'en tirer des déductions. C'est ce que l'on nomme le *moteur d'inférences* (cf. figure 2). Selon le type de système expert et le mode de construction de la base de connaissances (c'est-à-dire l'ensemble des règles), le logiciel peut éventuellement poser de "nouvelles" questions pour parfaire son diagnostic. En chaînage arrière, le système indique quelles conclusions il va chercher à établir, et quelles règles doivent être satisfaites pour y parvenir.

Dans tous les cas, on pourra connaître l'ensemble des variables définies, des prémisses établies, ainsi que le cheminement emprunté (activation ou non activation des règles).

La puissance et l'ordre

Les SE se classent selon leur capacité à traiter des propositions ou des variables ou à tolérer la présence de méta-règles activant ou inactivant les règles de premier niveau. L'ordre "zéro" ne traite que des règles simples (propositionnelles), l'ordre "zéro +" permet d'attribuer des coefficients d'importance aux règles, ce qui offre la possibilité d'en désactiver certaines en fonction des réponses apportées aux précédentes. Ce point est extrêmement important.

Prenons un exemple : soit à interroger un téléspectateur sur sa consommation télévisuelle hebdomadaire. Pour un habitué de la programmation, aucun problème, il suffit de placer un "Input" et d'aller à une étiquette en fonction de la réponse (exprimée en heures). Avec un SE d'ordre zéro, le logiciel va poser imperturbablement toutes les questions correspondant aux fourchettes préétablies, sans "comprendre" qu'une seule réponse suffit. Seule une méta-règle permettrait de lui indiquer de sauter les questions suivantes aussitôt qu'une réponse positive a été donnée.

De ce point de vue un "interfaçage" avec un langage de programmation simple constituerait un "plus" très important permettant de se passer de la lourdeur des systèmes d'ordre 1 ou 1+ tout en disposant d'une puissance de traitement presque comparable pour des sujets pas trop difficiles à formaliser.

Ajoutons enfin que la constitution de la base de connaissance exige une très grande rigueur dans la syntaxe (et l'orthographe) des règles. En effet, les SE ne sont pas encore capables de faire de la reconnaissance "intelligente" (par analyse morpho-syntaxique), ce qui oblige l'expert (ou le cogniticien) à *présenter les règles sous une forme répétitive et stricte*, un peu comme avec la syntaxe d'un langage de programmation, mais avec l'inconvénient que jamais le SE ne renvoie un bon vieux "Syntax error". Si une règle est définie avec un qualificatif précis, il est indispensable de le conserver à l'identique dans toute la base, sans la moindre variation, sous peine de voir s'écrouler tout le raisonnement (et de passer quelques heures à trouver la source du parasitage). Les esprits rapides auront tout de suite imaginé la parade : le traitement de texte... En effet, les fonctions de recherche et de remplacement, de copier/coller permettent de traiter facilement les règles et de leur assurer une homogénéité parfaite. Malheureusement, la quasi-totalité des traitements de texte n'enregistrent pas les règles en mode "texte", mais selon des standards internes... Seul (ou presque), Turbo Expert le permet.

Sa rusticité lui confère sa puissance.

Les points forts de Turbo Expert

En premier lieu, il faut insister sur son prix très attractif pour ce genre de logiciels. Les tarifs pratiqués par les autres éditeurs montrent bien qu'il s'agit encore d'un marché professionnel excessivement restreint, réservé aux passionnés travaillant dans des entreprises fortunées.

Turbo Expert est donc un SE d'ordre zéro (le plus simple), mais qui permet néanmoins de développer des bases relativement complexes. Comme indiqué plus haut, les règles et les bases de faits (c'est-à-dire la série de questions auxquelles les réponses ont été positives) *sont enregistrées en mode texte*, ce qui permet de les créer et de les modifier dans son traitement de texte favori, à condition de ne pas placer le moindre signe de ponctuation, lequel parasite tout (sur la version testée, n°2.3.2.). En début de fichier on doit simplement indiquer, derrière la lettre "R" le nombre de règles, et devant chacune d'elles, le nombre de lignes. Il est ainsi possible de construire ses règles en restant dans le traitement de texte et en copiant/collant à chaque fois que nécessaire ou en recourant à un glossaire.

Avec Multifinder (ou avec un accessoire du genre de MockWrite, présentant un traitement de texte dans le menu "pomme"), on peut activer Turbo Expert, se livrer à des expertises d'essai et revenir dans le traitement de texte pour effectuer les (nombreuses) mises au point indispensables à ce genre d'activité.

Turbo Expert indique le **dictionnaire des prémisses** (sur d'autres logiciels, ce sera la liste des variables). Sur la figure 3, nous donnons un exemple réel d'une base de près de 200 règles sur une typologie des associations françaises, obtenues à partir d'une soixantaine d'interrogations. La dernière ligne constitue toujours la conclusion.

On peut évidemment ajouter ou supprimer des règles depuis Turbo-Expert, mais nous avons préféré opérer dans le traitement de texte (en l'occurrence Word 3) parce qu'il nous a semblé qu'une lecture continue était plus pratique qu'une lecture séquentielle (ne présentant à l'écran qu'une seule règle à la fois).

Écrit en Turbo Pascal, Turbo Expert est très rapide, mais, avec 200 règles, il n'est pas instantané. On a l'impression qu'il "réfléchit" (cf. figure 4). Il signale les règles qui ont été activées.

On peut enregistrer la base de faits (c'est-à-dire l'ensemble des assertions vérifiées) et la réutiliser pour la mise au point, ce qui est bien pratique et évite de devoir répondre à chaque fois à plusieurs dizaines de questions. Avec un peu d'astuce, on arrive ainsi assez rapidement à enregistrer plusieurs bases de faits légèrement différentes, de façon à chercher à *piéger* son SE. Il est possible de stopper une interrogation en cours et de demander la déduction correspondante, puis de reprendre la chaîne des déductions si aucune conclusion n'a pu être tirée.

En chaînage arrière, le système donne la liste des conclusions, demande laquelle doit être vérifiée et indique ensuite l'ensemble des règles concernées qui doivent l'être à leur tour. Il propose alors de repasser en mode déduction (chaînage avant). L'induction est très pratique pour la mise au point du SE, on peut ainsi contrôler le "degré d'activation" de ses hypothèses.

Les points faibles

Nous avons déjà signalé les difficultés d'affichage, l'éloignement de l'ergonomie propre au Macintosh, les difficultés du copier/coller. Ce sont là des défauts de transcription (Turbo SE a été développé dans le monde MS-DOS, on s'en rend compte immédiatement, mais ils pourraient facilement être gommés si un marché suffisant s'offrait à l'auteur.

Il est d'autres défauts inhérents à l'ordre zéro lui-même, en particulier l'absence de métarègles et l'impossibilité de programmer les interrogations de manière dynamique

(en fonction des réponses aux questions précédentes). Peut-être peut-on espérer une version 0+ dans un avenir pas trop lointain.

A cet effet, voici comment il demeure possible de s'affranchir de l'absence de coefficients de vraisemblance (et dans une certaine mesure de construire des bases plus homogènes). Il suffit de prévoir *une gradation très nette dans les conclusions* et de reprendre ces conclusions en effet cumulatif dans de nouvelles règles (cf. figure 5). Avec un peu d'attention, on arrive assez bien et assez vite à des modélisations satisfaisantes, réellement utilisables par des non-experts.

Conclusion

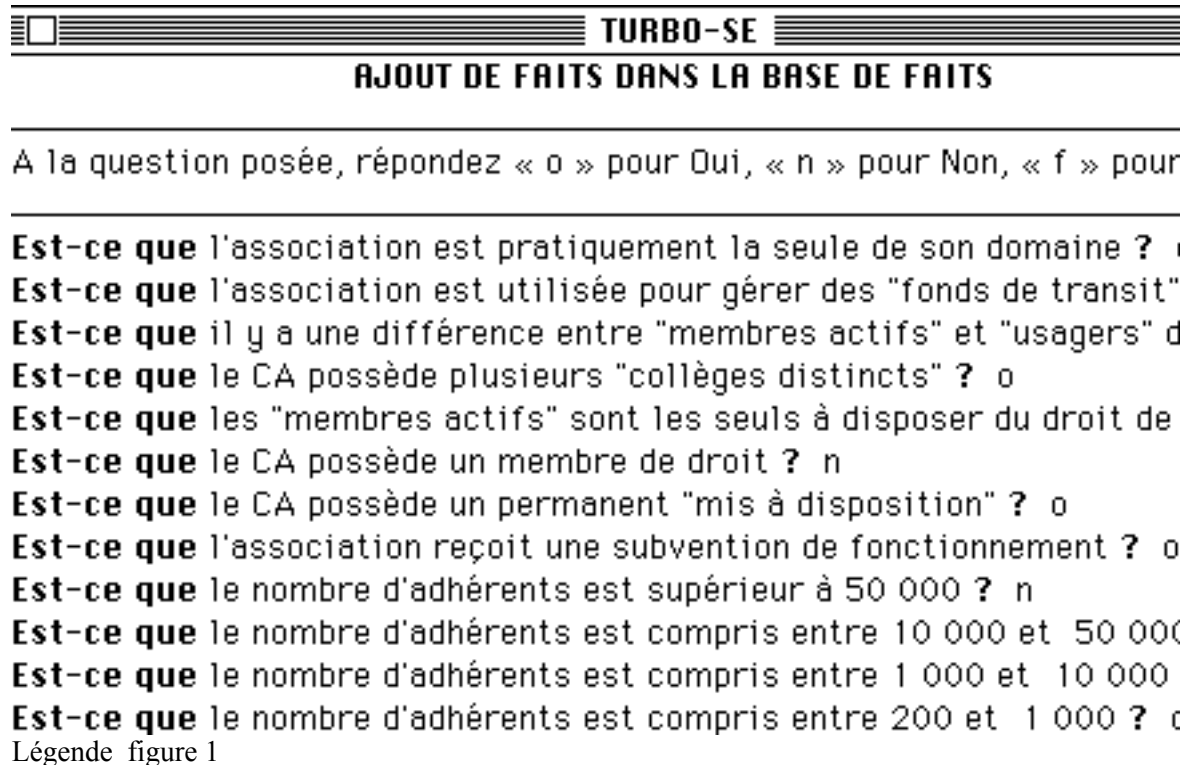
En conclusion, un excellent produit, malgré quelques défauts de finition, pouvant donner satisfaction non seulement à des personnes simplement désireuses de découvrir l'architecture générale d'un système expert mais aussi à des experts susceptibles d'organiser ou de formaliser leur connaissance. Ajoutons que dans le guide d'emploi (figurant sur la disquette programme), l'auteur, Philippe Larvet, explique avec force détails le fonctionnement des systèmes experts tout en donnant quelques exemples d'utilisation.

De plus, la fréquentation de Turbo Expert rendra plus exigeant vis-à-vis des autres SE plus puissants et beaucoup plus chers.

Turbo Expert en bref

- Système expert d'ordre zéro (sans métarègles).
- Jusqu'à 500 règles (version 2.3.2.)
- Jusqu'à six prémisses par règle.
- Une seule conclusion par règle (il faut redéfinir des conclusions comme prémisses).
 - Base des faits (des assertions) enregistrable en mode texte et directement expertisable.
 - Base des connaissances (des règles) en mode texte, permettant une mise au point facile dans un traitement de texte standard (portabilité et bons transferts).
 - Chaînage avant et arrière (déduction, induction).
 - Dictionnaire des prémisses.
 - Fonctions classiques des systèmes experts (ajout, suppression de règles).
 - Translation de règles possible par l'intermédiaire du traitement de texte.
 - Possibilité (théorique) de récupérer ses bases de connaissances dans le cas d'un passage sur de plus gros logiciels.
- Pas de gestion correcte des affichages (ascenseurs, cases d'agrandissement, etc.).
- Pas d'analyseur de syntaxe. Impossibilité de ponctuer les règles.

Figure 1 :



Comme on peut le constater, le SE ignore la syntaxe ("est-ce que il y a..."). Les réponses sont de simples oui/non, et on est loin d'un dialogue en langage naturel.

Figure 2 :

32 Faits chargés en mémoire.

>> Déductions effectuées (en Chaînage avant) :
(Règle 1): l'association pourrait être une association-alibi
(Règle 150): l'association est sûrement une association-alibi
(Règle 151): Conclusion assez innovante et indépendante
(Règle 152): Conclusion innovante et assez indépendante
(Règle 168): CONCLUSION innovante indépendante et alibi
(Règle 172): CONCLUSION indépendante et alibi
>>>> Fin de déduction <<<<<

Légende figure 2 :

On remarque l'emploi des gradations (de "pourrait être" à "est sûrement"). En fin de déduction, le système donne lui-même ses conclusions. On peut évidemment ajouter des règles débouchant sur le doute, en particulier lorsqu'il y a des contradictions dans les réponses ou lorsque l'expertise est très complexe.

Figure 3 :

R
174
2
l'association est pratiquement la seule de son domaine
l'association pourrait être une association-alibi
2
l'association est utilisée pour gérer des "fonds de transit"
l'association est sûrement une association-relais !
3
il y a une différence entre "membres actifs" et "usagers" de l'association
le CA possède plusieurs "collèges distincts"
l'association pourrait être une association-relais

Légende Figure 3

Quelques-unes des règles de la base de connaissance. Le repérage du nombre exact de règles n'est pas toujours facile, en particulier avec un grand nombre de lignes. On peut très bien employer Excel et faire une somme sur les résultats de la fonction VAL (qui prendra la valeur 1 pour toutes les grandeurs numériques, en l'occurrence les nombres de lignes de chaque règle. Essayez, ça marche !

Figure 4 :

14 Faits chargés en mémoire.

>> Déductions effectuées (en Chaînage avant) :

(Règle 8): l'association pourrait être une association-relais
(Règle 33): l'association est peut-être une association-relais
(Règle 34): l'association est sûrement une association-relais
(Règle 35): l'association est une association-relais
(Règle 58): l'association pourrait être une association-alibi
(Règle 113): l'association est sûrement innovante
(Règle 134): l'association pourrait être innovante
(Règle 154): Conclusion relais et assez innovante
(Règle 155): Tendence assez innovante et sûrement relais
(Règle 18): l'association est peut-être une association-alibi
(Règle 150): l'association est sûrement une association-alibi
(Règle 169): CONCLUSION relais et alibi peu d'utilité sociale
>>>> Fin de déduction <<<<

Légende Figure 4 :

On observe ici une déduction plus complexe. Le SE manipule plusieurs hypothèses, revient en cours de route à une règle antérieure (la n°18) et finit par livrer une déduction.

Figure 5 :

5 l'association reçoit une subvention de fonctionnement le nombre d'adhérents est inférieur à 100 personnes l'association pourrait être une association-relais l'association est peut-être une association-relais l'association est sûrement une association-relais
--

Légende Figure 5 :

On voit que le système combine des faits bruts (les deux premières lignes) avec des conclusions déjà tirées ailleurs. S'il observe une accumulation de conclusions concomitantes ("pourrait être", "est peut-être"), il dégage une préconclusion ("est sûrement"). Selon la complexité de la modélisation, on peut ainsi enchaîner jusqu'à six niveaux de prémisses pré-conclusives, ce qui permet de formaliser des cas assez complexes.

450 F TTC, franco, non protégé, avec mode d'emploi et exemples. Les établissements d'enseignement peuvent aussi obtenir une licence d'utilisation pour 800 F TTC (franco). Ph. Larvet livre alors le code source de son programme et autorise les copies à usage pédagogique.

Pour recevoir ce programme, envoyer un chèque de 450 F à l'auteur, Philippe LARVET, Le Buc, Sérignac Péboudou - 47410 LAUZUN.

Voir aussi le très intéressant ouvrage du même auteur, *Systèmes Experts en Turbo Pascal*, Paris, Editions Eyrolles, 1987.

Dernière minute !

Ph. Larvet annonce un logiciel d'ordre zéro + pour le début 1988. Nous en reparlerons.