Visualisation exploratoire des résultats d'algorithmes d'arbre de décision

Thanh-Nghi Do*, Nguyen-Khang Pham**, François Poulet***

*Equipe InSitu, INRIA Futurs, LRI, Bat.490, Université Paris Sud 91405 Orsay Cedex
Thanh-Nghi.Do@lri.fr
http://www.lri.fr/~dtnghi
**Equipe Texmex, IRISA, 35042 Rennes Cedex
pnguyenk@irisa.fr
***ESIEA-Ouest, 38, rue des Docteurs Calmette et Guérin, 53000 Laval
francois.poulet@esiea-ouest.fr
http://visu.egc.free.fr

Résumé. Nous présentons une méthode d'exploration des résultats des algorithmes d'apprentissage par arbre de décision (comme C4.5). La méthode présentée utilise simultanément une visualisation radiale, focus+context, fisheye et hiérarchique pour la représentation et l'exploration des résultats des algorithmes d'arbre de décision. L'utilisateur peut ainsi extraire facilement des règles d'induction et élaguer l'arbre obtenu dans une phase de post-traitement. Cela lui permet d'avoir une meilleure compréhension des résultats obtenus. Les résultats des tests numériques avec des ensembles de données réelles montrent que la méthode proposée permet une bien meilleure compréhension des résultats des arbres de décision.

1 Introduction

Le volume de données stocké double actuellement tous les 9 mois (Lyman et al, 2003) et donc le besoin d'extraction de connaissances dans les grandes bases de données est de plus en plus important (Fayyad et al, 2004). La fouille de données (Fayyad et al, 1996) vise à traiter des ensembles de données pour identifier des connaissances nouvelles, valides, potentiellement utilisables et compréhensibles. Cette utilisabilité est fonction des buts de l'utilisateur donc seul l'utilisateur peut déterminer si les connaissances extraites répondent à ses attentes. Les outils de fouille de données doivent donc être interactifs et anthropocentrés. Notre approche consiste à impliquer plus fortement l'utilisateur dans le processus de fouille par des méthodes graphiques interactives dans un environnement de fouille.

De nombreuses méthodes de visualisation ont été développées dans différents domaines et utilisées pour l'analyse exploratoire et la fouille de données (Fayyad et al, 2001), (Keim, 2002). Les méthodes de visualisation peuvent être utilisées pour le pré-traitement de données (par exemple la sélection de données) ou en post-traitement (par exemple pour voir les résultats). Des méthodes récentes (Ankerst, 2001), (Do et Poulet, 2004a et b), (Munzner, 1997) essayent d'impliquer plus significativement l'utilisateur dans le processus de fouille de

données par le biais de la visualisation. Parmi les avantages que ce type d'approche présente on peut citer : l'utilisateur peut être un spécialiste du domaine des données et utiliser son expertise tout au long du processus de fouille, la confiance et la compréhensibilité du modèle sont accrues car l'utilisateur a participé à sa construction et enfin on peut utiliser les performances de la perception visuelle humaine en reconnaissance des formes.

Nous présentons une méthode d'exploration interactive des résultats des algorithmes d'arbres de décision comme C4.5 (Quinlan, 1993). Les arbres de décision sont des méthodes efficaces et populaires (Kdnuggets, 2003 et 2004) pour la classification. Un arbre de décision est un classifieur représenté sous la forme d'un arbre dans lequel chaque nœud est soit une feuille contenant l'étiquette de la classe, soit un nœud interne contenant un test à effectuer sur l'un des attributs avec un fils pour chaque résultat possible du test. Une règle d'induction (de la forme si alors) est créée pour chaque chemin partant de la racine de l'arbre et parcourant les tests (en faisant des conjonctions) jusqu'à la feuille qui est l'étiquette de la classe. Les arbres de décision sont particulièrement appréciés car ils permettent une compréhension aisée, mais lors d'une tâche de classification relativement complexe l'utilisateur n'est plus capable d'explorer efficacement les résultats obtenus sous forme textuelle.

Quelques techniques de visualisation d'arbres de décision existent déjà. Le Tree-Visualizer de Mineset (Brunk et al, 1997) permet une visualisation en 3D avec une vue partielle des résultats. Seuls les premiers niveaux de l'arbre sont affichés initialement et le reste apparaît au fur et à mesure que l'utilisateur descend dans l'arbre.

Le Cone Tree (Robertson et al, 1991) utilise aussi une représentation 3D de l'information hiérarchique, le nœud courant est le sommet d'un cône contenant tous ses fils répartis à la base de ce cône 3D.

La technique focus+context (Lampig et Rao, 1996) utilise une portion de l'espace plus importante pour la zone où l'utilisateur se positionne. Le principe de cette visualisation est l'utilisation d'un plan hyperbolique et d'une projection sur une sphère.

Les arbres hyperboliques (hyperbolic trees (Munzner, 1997)) visualisent les arbres sous la forme de graphes dans un espace 3D hyperbolique. Cette technique utilise une métrique hyperbolique et optimise le placement des fils d'un nœud sur une hemisphère autour de la base du cône. L'exploration animée (Yee et al, 2001) de graphes dynamiques avec visualisation radiale est une technique d'animation de la transition d'un nœud à l'autre. Pour permettre une transition facile à suivre, l'animation correspond à une interpolation linéaire entre les nœuds source et destination avec des contraintes sur l'ordre et l'orientation des objets.

En ce qui concerne les arbres de décision dans un contexte de fouille de données, les impératifs importants pour l'utilisateur sont : une compréhension facile, la possibilité d'extraire des règles pertinentes et la possibilité d'effectuer un élagage de l'arbre dans une étape de post-traitement. Nous présentons une nouvelle méthode de visualisation radiale des arbres de décision pour l'exploration interactive. Nous utilisons pour cela des méthodes de visualisation telles qu'une représentation à la manière d'un explorateur, le focus+context, le fisheye, la visualisation hiérarchique et les techniques interactives pour représenter des arbres de grandes tailles de manière graphique plus intuitive que les résultats habituels des algorithmes d'arbres de décision. L'utilisateur peut explorer interactivement l'arbre de décision, extraire des règles intéressantes et élaguer l'arbre obtenu. Les résultats numériques des tests sur des ensembles de données réels montrent que les méthodes proposées améliorent la compréhension des résultats des arbres de décision.

2 Représentation type explorateur

Une représentation type explorateur va projeter la racine de l'arbre dans le coin supérieur gauche de la vue (habituellement de coordonnées (0,0)). Un nœud est représenté par un carré dont la couleur représente la classe (ou classe majoritaire). La taille d'une feuille sera proportionnelle aux nombres d'individus contenus dans cette feuille. Chaque nœud peut être développé ou non par un clic de souris. La figure 1 est un exemple de visualisation de type explorateur avec l'ensemble de données Shuttle de l'UCI (Blake et Merz, 1998) contenant 58000 individus en dimensions 9 et 7 classes.

L'utilisateur peut ainsi explorer facilement les résultats de l'algorithme d'arbre de décision dans un mode graphique plus intuitif que les résultats habituels sous forme de texte. La représentation du nombre d'individus contenus dans les feuilles aide à déterminer l'importance de la règle de décision. Les feuilles ne contenant que peu d'individus sont généralement de moindre importance. Ces indications sont une aide précieuse lorsque l'utilisateur cherche à élaguer l'arbre de décision obtenu.

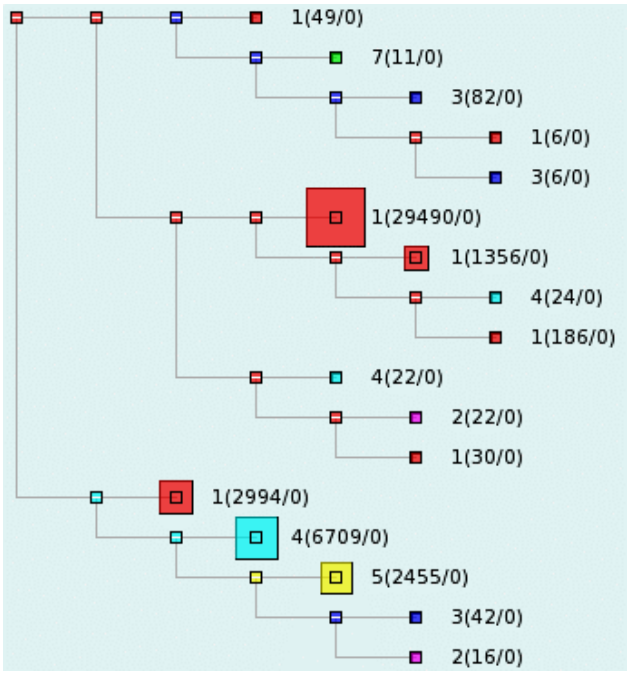

FIG. 1 – *Représentation type explorateur de l'arbre de décision des données Shuttle*

L'utilisateur peut extraire des règles d'induction, en cliquant sur une feuille de l'arbre, le chemin de la racine à la feuille est sélectionné et la règle correspondante est alors affichée. L'exemple de la figure 2 montre une règle extraite de l'arbre de décision obtenu sur l'ensemble de données Shuttle. Grâce aux techniques interactives telles que le développement ou non d'un nœud, le focus ou le zoom, l'utilisateur peut facilement naviguer dans l'arbre avec l'affichage des informations associées à chaque nœud telles que le nombre d'individus

ou le nombre d'erreurs (individus mal classifiés). Avec ces éléments l'utilisateur a une aide précieuse pour effectuer lui-même l'élagage de l'arbre de décision. Cet élagage permettra d'améliorer le taux de bonne classification sur l'ensemble de test en minimisant le sur-apprentissage.

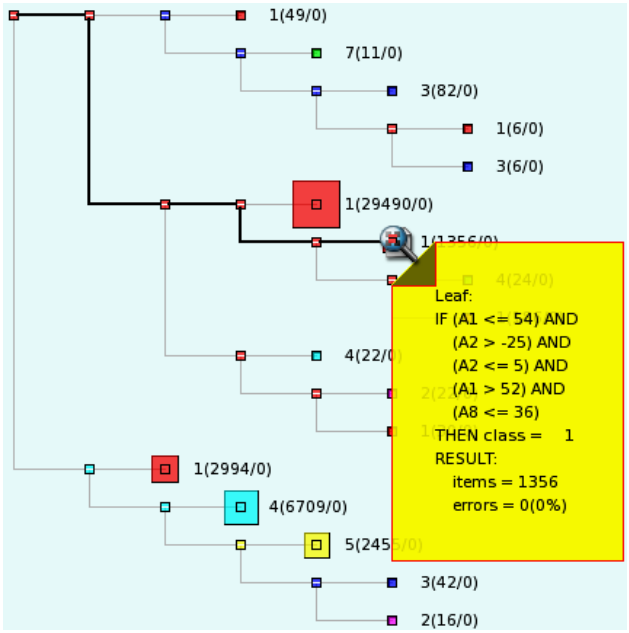

FIG. 2 – *Extraction de règle d'induction*

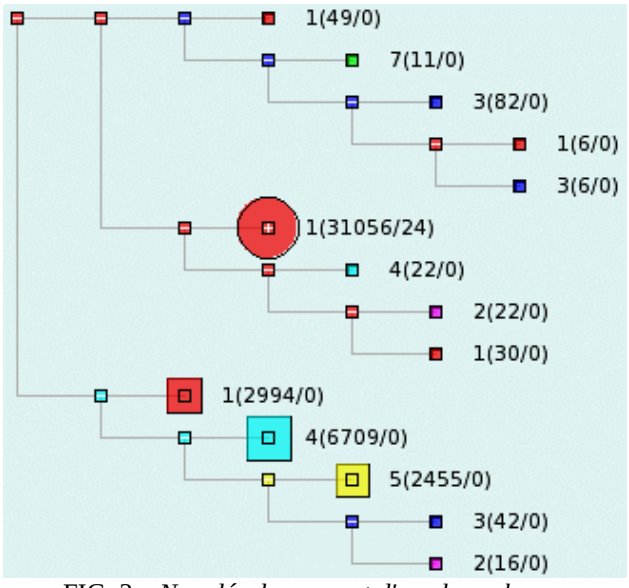

FIG. 3 – *Non développement d'une branche*

Sur l'exemple de la figure 3, l'utilisateur peut ne pas développer une branche de l'arbre de décision car il a l'information sur le nombre d'individus et d'erreurs concernés, la profondeur dans l'arbre et la classe. Il peut donc décider d'élaguer le sous-arbre correspondant comme montré sur la figure 4 car 24 individus sont de la classe 4 et 31032 de la classe 1 ce qui représente un pourcentage inférieur à 0,08% d'individus de la classe minoritaire. Ce type de méthode permet d'impliquer de manière plus significative l'utilisateur dans une phase de post-traitement par le biais de méthodes de visualisation interactives.

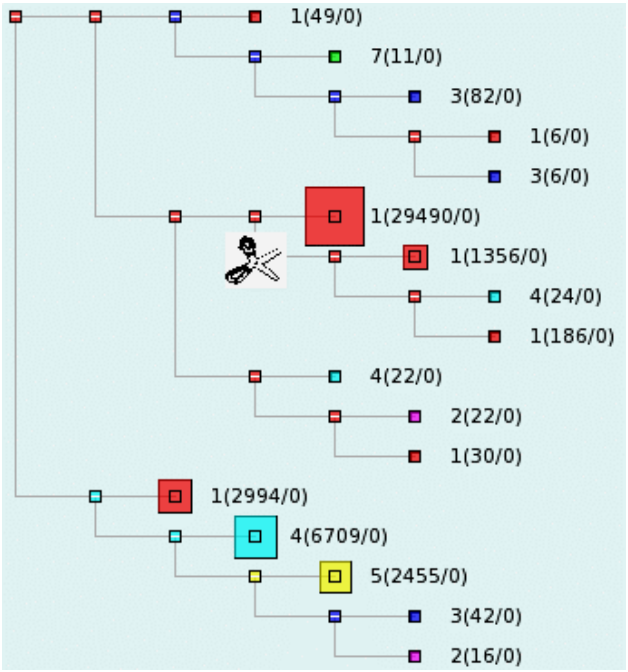

FIG. 4 – *Elagage de l'arbre de décision*

La représentation de type explorateur est performante pour des arbres de tailles moyennes avec à la fois une grande simplicité d'utilisation et une bonne compréhension par l'utilisateur.

3 Visualisation radiale, fisheye, focus+context et techniques hiérarchiques

Pour des arbres de grandes tailles nous avons essayé d'améliorer la représentation 2D en utilisant une visualisation radiale, le fisheye et le focus+context dans une visualisation hiérarchique. Avec l'algorithme de visualisation radiale, la racine de l'arbre de décision est projetée sur le centre de la vue, les nœuds fils sont disposés en cercles ou arcs de cercle autour du nœud parent. L'espacement est fonction du nombre de nœuds descendants de chaque nœud, comme le montre l'exemple de la figure 5.

Nous utilisons ensuite la technique du fisheye qui aide l'utilisateur à détailler l'environnement immédiat du nœud courant ce qui lui permet de se focaliser sur les parties qu'il juge intéressantes de l'arbre de décision.

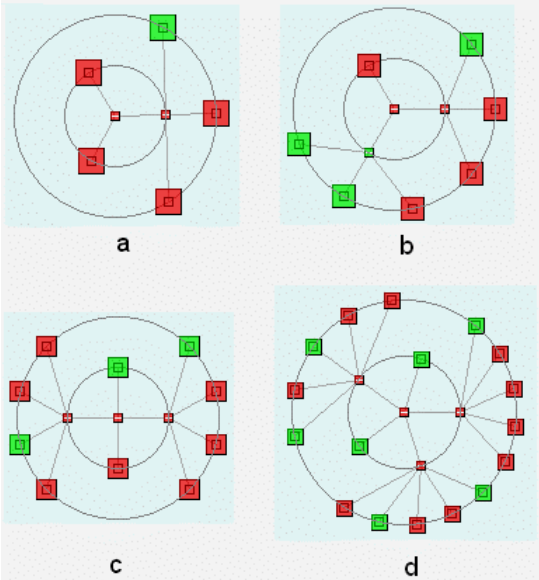

FIG. 5 – *Visualisation radiale*

Nous proposons aussi une nouvelle technique dérivée du fisheye et permettant un zoom sur une zone d'intérêt au détriment des autres parties. Considérons la racine O projetée en O1 et un nœud P projeté en P1, comme sur l'exemple de la figure 7. La racine va être translatée de O1 vers O2 en préservant l'angle α= (Ox, OP). Le noeud P doit donc lui aussi être translaté de P1 à P2. L'angle obtenu (O1x, O1P2) est supérieur à l'angle (O1x, O1P1). Voici comment obtenir la position de P2 :

$P_2(x) = x_1 + r*\cos(\beta)$ (1)

$P_2(y) = y_1 + r*\sin(\beta)$ (2)

$P_2(x) = x_2 + t*\cos(\alpha)$ (3)

$P_2(y) = y_2 + t*\sin(\alpha)$ (4)

En substituant P2(x) [resp. P2(y)] dans (1) [resp.(2)], on obtient les équations (5) et (6).

$r*\cos(\beta) = x_2 - x_1 + t*\cos(\alpha)$ (5)

$r*\sin(\beta) = y_2 - y_1 + t*\sin(\alpha)$ (6)

$\Rightarrow r^2 = (\Delta x + t*\cos(\alpha))^2 + (\Delta y + t*\sin(\alpha))^2$ (7)

avec $\Delta x = x_2 - x_1$ et $\Delta y = y_2 - y_1$

$t^2 + 2(\Delta x \cos(\alpha) + \Delta y \sin(\alpha))t + \Delta^2 x + \Delta^2 y - r^2 = 0$ (8)

On ne conserve que la plus grande t* des solutions de (8) et la nouvelle position de P2 est obtenue en substituant t dans (3) et (4).

L'utilisateur peut ainsi se focaliser sur les zones d'intérêt par sélection à la souris.

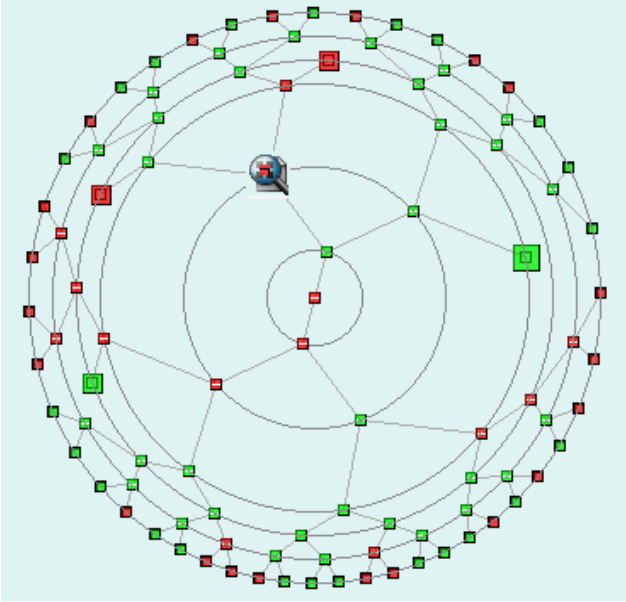

FIG. 6 – *Visualisation radiale et fisheye*

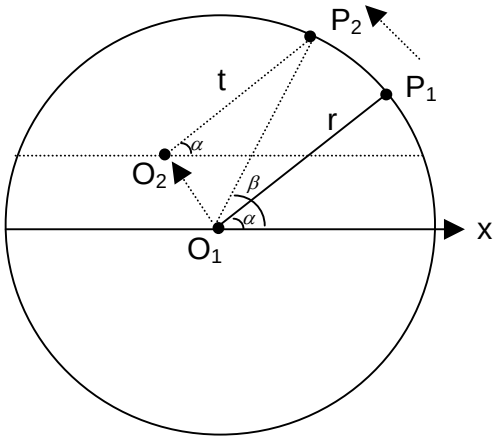

FIG. 7 – *Focus sur les nœuds d'intérêt*

Les expérimentations des systèmes de visualisation d'arbres de décision (Barlow et Neuville, 2001) ont montré qu'il n'y a pas une technique meilleure que les autres pour l'exploration de grands arbres de décision avec à la fois la simplicité, la vitesse, le focus, une vue globale et une bonne compréhension de l'utilisateur. Nous souhaitons donc combiner plusieurs techniques pour tirer partie des avantages de chacune. Notre méthode de visualisation utilise une visualisation type explorateur, une visualisation radiale, le fisheye et le focus. Le principe est de diviser l'espace écran en différentes zones dans lesquelles une visualisation différente peut être affichée. Nous avons choisi par défaut de visualiser l'arbre

complet avec la visualisation radiale, le fisheye et le focus. L'utilisateur peut sélectionner un sous-ensemble de l'arbre et le visualiser avec une autre méthode comme la représentation type explorateur sur l'exemple de la figure 9.

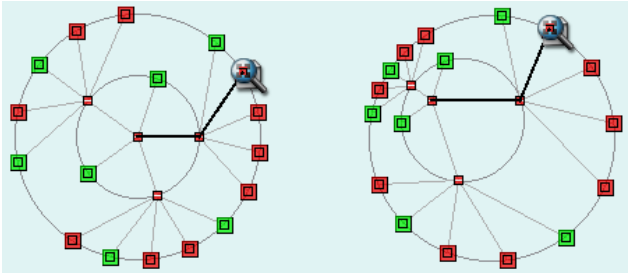

FIG. 8 – *Focus sur une région d'intérêt*

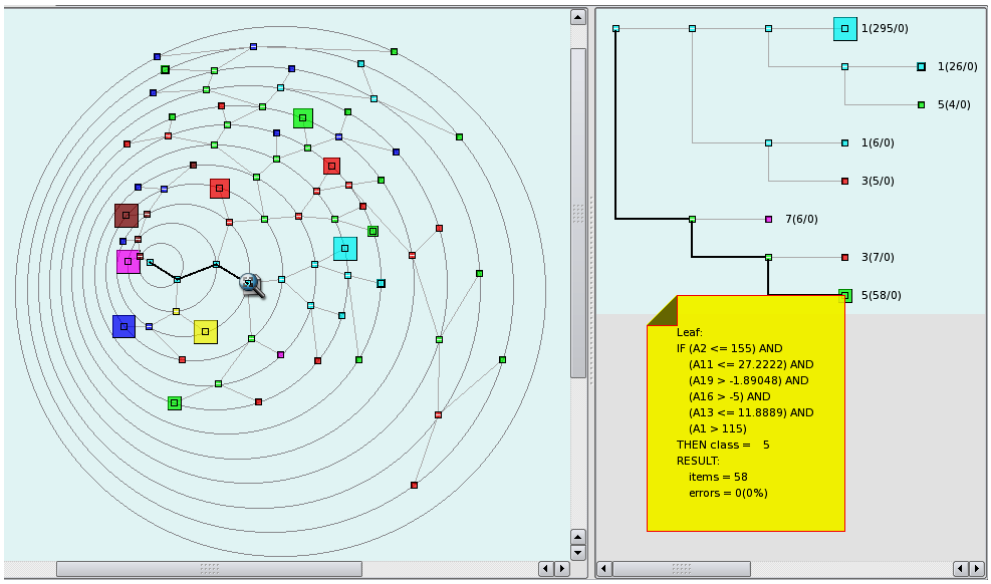

FIG. 9 – *Visualisation hiérarchique et type explorateur de Segment*

La visualisation hiérarchique permet de préserver une vision globale de l'arbre avec une représentation plus fine grâce au fisheye et au focus. Le sous-arbre sélectionné peut alors être exploré aisément dans la représentation type explorateur. On allie donc la simplicité, la rapidité pour effectuer la tâche, la facilité d'utilisation et la compréhension du modèle. La figure 9 représente la visualisation de l'arbre de décision obtenu sur l'ensemble de données Segment de l'UCI (2310 individus en dimension 19 et 7 classes).

4 Exploration des résultats de la classification de spam

Tous les outils présentés dans cet article ont été développé en C/C++ sous Linux en utilisant la librairie open-source Qt. Nous avons aussi inclus l'algorithme d'induction d'arbre de décision C4.5.

Nous nous intéressons à la classification de spam (http://www.ics.uci.edu/~mlearn/databases/spambase). La base de données spam a été créée par G.Forman et ses collègues du laboratoire de Hewlett-Packard. C'est une collection de mails de leur postmaster et de leurs courriers personnels et donc le mot "george" et le code régional "650" sont des indicateurs de non spam. Ces informations sont utiles si l'on désire créer un filtre antispam, mais d'autres préfèreront les cacher pour créer un filtre anti spam plus générique. La base de données spam contient 4601 individus (1813 spam, soit 40%) et 58 dimensions. La dernière colonne est la classe qui indique si le mail est du spam ou non. La plupart des attributs indiquent si un mot ou caractère particulier a été rencontré fréquemment dans le mail. Il y a 48 attributs à valeurs réelles, 2 attributs mesurant la longueur des séquences de lettres majuscules consécutives et des mesures statistiques sur les attributs.

L'algorithme d'arbre de décision C4.5 a classifié l'ensemble de données avec un taux de précision de 92,24% sur l'ensemble de test (comportant 1534 individus). L'arbre résultant contient 148 nœuds et est représenté sur la figure 10. L'utilisateur peut alors explorer cet arbre avec les techniques interactives de développement ou non de nœud, le focus ou le zoom, il a aussi à disposition les informations associées au nœud courant comme le nombre d'individus ou d'erreurs.

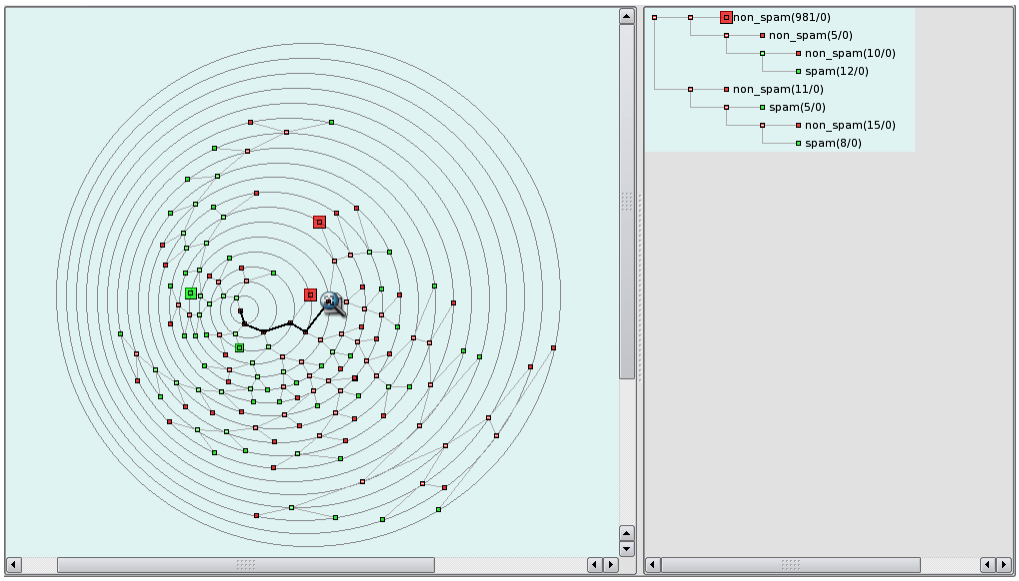

FIG. 10 – *Visualisation hiérarchique de l'arbre de décision de Spambase*

Tree-Viz

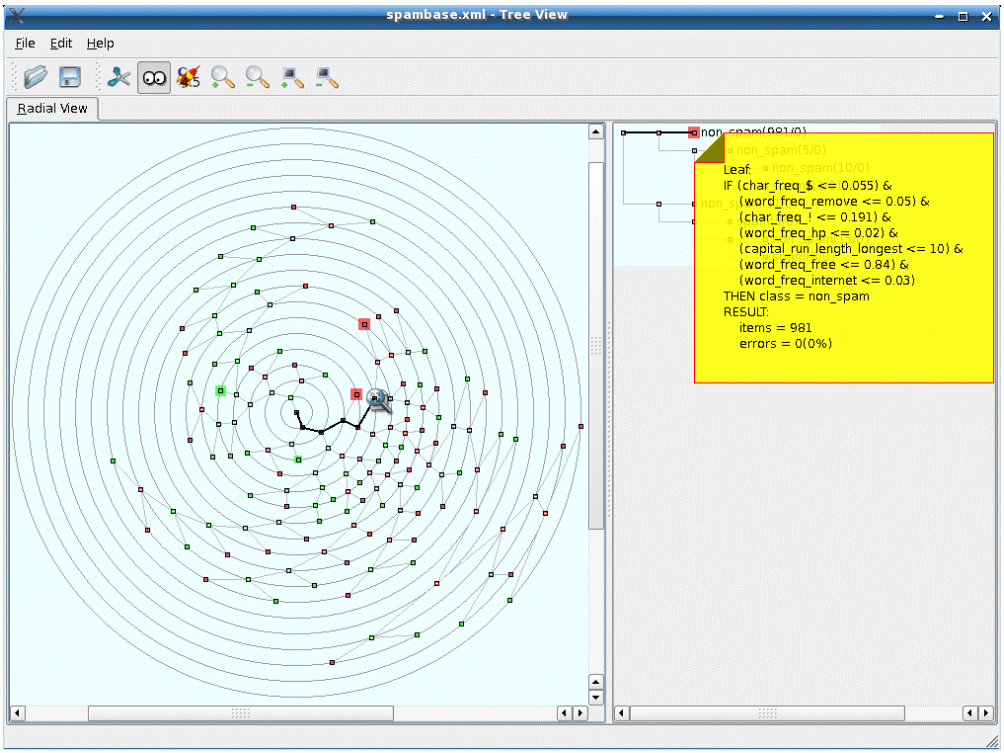

FIG. 11 – *Extraction de règles à partir de l'arbre de décision de Spambase*

Avec ces outils et les informations dont il dispose, l'utilisateur a une bonne connaissance pour extraire des règles d'induction ou élaguer interactivement l'arbre de décision, il peut se focaliser sur une région d'intérêt et explorer le sous-arbre correspondant à l'aide de la vue type explorateur. Sur les figures 10 et 11, on distingue par exemple des feuilles qui comportent une grande quantité d'individus (et de faible niveau de profondeur). On peut ainsi extraire 4 règles (2 pour la classe spam et 2 pour non spam) qui vont permettre de classifier de manière fiable la grande majorité de l'ensemble de données (l'une de ces règles est représentée en haut à droite de la figure 11). L'utilisateur a ainsi une bonne assimilation et compréhension des résultats de l'arbre de décision de l'ensemble de données Spambase.

L'environnement développé permet la visualisation d'arbre de taille relativement importante en permettant simultanément une vision de la totalité de l'arbre et un zoom sur les zones d'intérêt. Il essaye d'associer à la fois la hiérarchie complète de l'arbre, la simplicité d'utilisation, la rapidité d'exécution de la tâche et la satisfaction et bonne compréhension de l'utilisateur.

5 Conclusion et perspectives

Nous avons présenté un nouvel environnement graphique pour l'exploration des résultats des algorithmes d'arbre de décision (comme C4.5). Notre but était de satisfaire les demandes

des utilisateurs dans le contexte de la fouille de données : facilité d'interprétation, extraction de règles pertinentes et élagage efficace de l'arbre dans une phase de post-traitement. Nous avons développé une nouvelle méthode de visualisation radiale pour permettre l'exploration interactive des résultats des arbres de décision. Cette méthode a été utilisée en collaboration avec d'autres méthodes telles que la représentation type explorateur, le focus+context, le fisheye, la visualisation hiérarchique et les techniques interactives pour permettre la visualisation d'arbres de tailles importantes de manière plus intuitive que les habituelles sorties texte des algorithmes d'arbres de décision. Grâce à l'ensemble des techniques telles que le développement ou non des nœuds, le focus, le zoom ou la rotation l'utilisateur peut aisément explorer les résultats des arbres de décision. Il peut aussi prendre connaissance des informations associées aux nœuds de l'arbre comme le nombre d'individus ou d'erreurs. Cette connaissance peut lui permettre d'extraire des règles d'induction ou d'élaguer lui-même l'arbre dans un mode graphique interactif et intuitif basé sur la profondeur, la taille, la couleur et l'information liée aux nœuds de l'arbre. L'utilisateur a ainsi une meilleure compréhension de l'arbre obtenu. Les test sur l'ensemble de données Spambase montre que la méthode proposé permet de mieux appréhender les résultats des algorithmes d'arbres de décision. Une extension future est de trouver une abstraction permettant de visualiser des arbres de taille plus importante, une autre extension envisagée est d'étendre cette approche à d'autres algorithmes de fouilles de données pour permettre de d'expliquer les résultats obtenus par ces algorithmes.

Références

Ankerst, M. (2002). Report on the SIGKDD-2002 Panel The Perfect Data Mining Tool: Interactive or Automated. in *SIGKDD Explorations*, 4(2):110-111.

Barlow, T. and P. Neville (2001). A Comparison of 2-D Visualizations of Hierarchies. in *IEEE InfoVis*, 131-138.

Blake, C. and C. Merz (1998). UCI Repository of Machine Learning Databases. http:// www.ics.uci.edu/~mlearn/MLRepository.html.

Brunk, C., J. Kelly, and R. Kohavi (1997). MineSet: An Integrated System for Data Access, Visual Data Mining and Analytical Data Mining. proc. *KDD'97*, AAAI Press, 135-138.

Do, T-N. and F. Poulet (2004). Enhancing SVM with Visualization. *Discovery Science* 2004, E. Suzuki et S. Arikawa Eds., Lecture Notes in Artificial Intelligence 3245, Springer-Verlag, 183-194.

Do, T-N. and F. Poulet (2004). Cooperation between Visualization Methods and SVM Algorithms for Data Mining. in proc. of MCO'04, Computer Sciences, Modelling, Computation and Optimization in Information Systems and Management Sciences : Data Mining Theory, Systems and Applications, H.A. Le Thi et T. Pham Dinh Eds., Hermes Science, 569-576.

Fayyad, U., G. Piatetsky-Shapiro, and P. Smyth (1996). From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, 17(3):37-54.

Fayyad, U., G. Piatetsky-Shapiro, and R. Uthurusamy (2004). Summary from the KDD-03 Panel – Data Mining: The Next 10 Years. in *SIGKDD Explorations*, 5(2):191-196.

Fayyad, U., G. Grinstein, and A. Wierse (2001). *Information Visualization in Data Mining and Knowledge Discovery*. Morgan Kaufmann Publishers.

Kobsa A. (2004). User Experiments with Tree Visualization Systems. in *IEEE InfoVis*, 9-16.

Kdnuggets (2003). What data mining techniques you use regularly?. KDnuggets Polls, Nov 9 – 23, 2003. http://www.kdnuggets.com/polls/2003/data_mining_techniques.htm.

Kdnuggets (2004). Which data mining techniques you used in a successfully deployed application?. KDnuggets Polls, Sep 13-27, 2004. http://www.kdnuggets.com/polls/2004-/deployed_data_mining_techniques.htm.

Keim, D. (2002). Information Visualization and Visual Data Mining. in *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1-8.

Lamping, J. and R. Rao (1996). The Hyperbolic Browser: A Focus + Context Technique for Visualizing Large Hierarchies. in *Journal of Visual Languages & Computing*, 7(1):33-55.

Michie, D., D.J. Spiegelhalter, and C.C. Taylor (1994). *Machine Learning, Neural and Statistical Classification*. Ellis Horwood.

Munzner, T. (1997). H3: laying out large directed graphs in 3D hyperbolic space. in *IEEE InfoVis*, 2-10.

Pham, N-K. and T-N. Do (2006). Tree-View : post-traitement interactif pour des arbres de décision. Acte du 4ème Atelier Visualisation et extraction de connaissances, EGC'06, 6èmes Journées d'Extraction et Gestion des Connaissances, 103-110.

Poulet, F. (2003). Interactive Decision Tree Construction for Interval and Taxonomical Data. in proc. of VDM@ICDM'03, 3nd Workshop on Visual Data Mining, 183-194.

Quinlan, J. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers.

Robertson, G-G., J-D. Mackinlay, and S-K. Card (1991). Cone Trees: animated 3D visualizations of hierarchical information. in proc. of the *SIGCHI: ACM Special Interest Group on Computer-Human Interaction*, 189-194.

Yee, K-P., D. Fisher, R. Dhamija, and M. Hearst (2001). Animated Exploration of Dynamic Graphs with Radial Layout. in *IEEE InfoVis*, 43-50.

Summary

Our investigation in this paper aims at interactively exploring the decision tree results obtained by the machine-learning algorithm like C4.5. We propose a new graphical radial tree layout method for supporting interactive exploration of decision trees. A new interactive graphical toolkit has been developed using explorer-like, radial layout, focus+context, fisheye, hierarchical visualization and interactive techniques to represent large decision trees in a graphical mode more intuitive than the results in output of the C4.5 algorithm. The user can easily extract inductive rules and prune the tree in the post-processing stage. He has a better understanding of the obtained decision tree models. The numerical test results with real datasets show that the proposed methods have given an insight into decision tree results.