WP4_D1 : T0 + 12 (N. Ragot)

Etat de l'art en numérisation cognitive

L'objectif d'une partie du projet DIGIDOC est d'embarquer dans les scanners une brique logicielle leur conférant une certaine intelligence, une capacité à s'adapter au flux de traitement courant, voire au document en cours pour améliorer la qualité de la numérisation. Ces notions d'intelligence et d'amélioration contextuelle de la qualité de numérisation sont suffisamment vagues pour permettre différentes interprétations. Ainsi, il n'est pas rare de trouver dans les plaquettes marketing le terme « scanner intelligent ». C'est pourquoi, dans la première partie de ce document, nous allons présenter brièvement les propriétés de certains scanners actuels, notamment en ce qui concerne leur aptitude à gérer différents types de documents, en particulier ceux présentant des dégradations et donc à « s'adapter » automatiquement. Dans une seconde partie, nous verrons plus précisément ce que nous entendons par numérisation cognitive ou scanner cognitif (ou intelligent) au sein du projet DIGIDOC et nous présenterons un état de l'art des méthodes scientifiques qui devraient nous permettre de traiter cette problématique.

Les scanners intelligents sur le marché

La plupart des scanners actuels, qu'ils soient destinés à un large public ou bien professionnels, peuvent parfois être qualifiés de « scanner intelligent ». Cette appellation, à fort impact commercial, n'a pas de définition précise et regroupe un ensemble de fonctionnalités relatives à la fois aux avancées matérielles des scanners mais aussi et surtout aux suites logicielles qui leur sont associées. Le tableau ci-dessous donne quelques exemples de ces scanners aux fonctionnalités avancées. L'objectif n'est pas ici d'être exhaustif mais plutôt de fournir un échantillon représentatif de ce qui se fait à l'heure actuelle autour de l'aptitude du scanner à s'adapter au document.

Scanner	Technologies matérielles	Technologies logicielles	lien
Panasonic KV-S5055C (2011)	*Toughfeed* : repère automatiquement les agrafes sur les documents et détecte le passage simultané de deux feuilles pour stopper la numérisation en cours	Image Capture Plus : pour l'édition des pages scannées. Permet de : supprimer les éventuelles pages blanches ; de modifier l'ordre des pages ; de recadrer les images ; d'effacer automatiquement les traces laissées par d'éventuels trous sur le document original	http://www.panasonic.fr/html/fr_FR/Produits/Solutions+Bureautiques%3A+Scanners+et+Tableaux/Panasonic+lance+le+KV-S5055C/7940598/index.html

	Peut numériser simultanément des documents de différentes tailles et de différentes épaisseurs depuis le chargeur de documents		
i1320 de Kodak (2006)	scanne les deux côtés d'une feuille en un seul passage (2 CCD)	Perfect Page Scanning; iThresholding; adaptive threshold processing; deskew; autocrop; relative cropping; aggressive cropping; electronic color dropout; dual stream scanning; interactive color, brightness and contrast adjustment; automatic orientation; automatic color detection; background color smoothing; image edge fill; image merge; content based blank page detection; streak filtering; image hole fill; sharpness filter	http://fr.ubergizmo.com/2006/07/scanner-intelligent-i1320-de-kodak/ http://www.4capture.fr/kodak.php
IRISPen Express		Inclus un OCR (128 langues) : Lecture d'images floues et déformées ; Technologie de suivi de ligne unique ; gestion de la couleur du texte et du fond ; pratiquement n'importe quelle police, dans un large éventail de styles et de tailles (de 8 à 20 points), ainsi que des caractères spéciaux, des marques de lecture, des symboles spéciaux et des éléments de mise en forme, tels que les barres verticales séparant les cellules d'un tableau) ; Alphabets mixtes ; Traduction, synthèse vocale, reconnaissance de codes barre	http://www.irislink.com/c4-1692-225/IRISPen-6--Overview.aspx
I2S Digibooks	prévisualisation en temps réel, détection de format, auto-focus, temps d'exposition,	redressement, correction de courbure, effacement des doigts, découpage des pages, accentuation des détails, accentuation du contraste,	http://www.i2s-bookscanner.com/

	correction d'éclairage,	étalement des niveaux, courbe en S, => bookrestorer	
I2S Copybooks			

Pour résumer, de nombreuses avancées techniques ont été faites sur le plan matériel et notamment sur la capacité du scanner à traiter une quantité importante de documents tout en gérant différents formats. Ces fonctionnalités sont souvent couplées à des systèmes permettant la détection de problèmes mécaniques au plus tôt pour éviter les blocages. Bien que ces avancées soient nécessaires, elles ne concernent pas directement le projet DIGIDOC qui est centré sur la numérisation elle-même.

Sur cet aspect, certains scanners avancés permettent de régler automatiquement plusieurs paramètres en fonction du document. Ils s'appuient pour cela sur des technologies que l'on retrouve fréquemment dans le domaine de l'optique ou de la vision pour le calibrage de capteurs. Plus le scanner est avancé et plus cette mise au point peut se faire sur un nombre de paramètres importants (cf. Digibooks). Ce paramétrage automatique, tout à fait intéressant dans notre contexte n'en reste pas moins limité pour deux raisons. La première est qu'il nécessite souvent une intervention de l'opérateur pour vérifier voire corriger le calibrage, notamment en fonction du résultat souhaité. Cela est particulièrement vrai lorsqu'il s'agit de numériser des documents pour lesquels on souhaite une grande fidélité (patrimoine en particulier). La seconde raison, qui résulte en partie de la première, est que le temps nécessaire au calibrage est souvent conséquent. Il ne peut donc se faire que ponctuellement.

Bien que cela n'apparaisse pas explicitement dans le tableau ci-dessus, les scanners possèdent aussi souvent des profils standards de numérisation qui peuvent être choisis par l'utilisateur, voire automatiquement par un pré scan. Ces profils permettent d'adapter notamment la résolution et la couleur en fonction du contenu du document (image, image et texte, texte seul) afin de trouver un compromis entre la taille des images acquises et l'information qu'elles contiennent. Là encore, bien que cette notion de profil corresponde à ce que l'on souhaiterait avoir dans un scanner intelligent, leur nombre reste très limité et ceux-ci ne dépendent absolument pas de l'usage que l'on souhaite faire du document numérique.

Une grande majorité (pour ne pas dire la quasi totalité) de ces scanners embarquent également des outils logiciels qui permettent « d'améliorer la numérisation » en corrigeant les défauts principaux (recadrage, suppression de pages blanches, amélioration du contraste, suppression du « bruit » ou de défauts (trous, doigts, etc.), etc.). Il faut bien comprendre ici que cette « amélioration de la numérisation » constitue un abus de langage dans le sens où la numérisation n'est en rien changée. Seul le résultat de la numérisation, c'est-à-dire l'image, est amélioré, dans le but satisfaire au mieux à des critères visuels subjectifs de l'utilisateur standard. Ces améliorations sont le plus souvent standards et ne correspondent bien souvent qu'à un usage de consultation du document pour un utilisateur quelconque. La seule exception

concerne l'ensemble des traitements, là encore standards, qui serviront à améliorer, outre le rendu de la numérisation, les performances d'un OCR.

Outils scientifiques pour rendre un scanner « intelligent »

Dans le projet DIGIDOC, nous souhaitons étendre les capacités des scanners afin d'adapter automatiquement le processus de numérisation en fonction du contexte de numérisation. Ce contexte représente non seulement le document en cours de traitement mais également l'ensemble des documents scannés au préalable (mémoire du flux), ainsi que l'usage qui sera fait des documents numérisés. L'usage doit permettre de déterminer le plus automatiquement possible les meilleurs paramètres de numérisation pour un flux. Ensuite, la gestion du contexte doit permettre de détecter les dérives du réglage ou encore les documents avec des spécificités telles qu'il faut ponctuellement adapter les paramètres. Le problème revient donc à trouver un jeu de paramètres de numérisation adapté à un usage et à un document (un profil ou protocole de numérisation). Même si certains profils types peuvent être établis *a priori*, la plupart pourront varier en fonction des utilisateurs, des documents, etc. Il faut donc que le scanner soit capable de se construire de façon incrémentale, au cours du temps, une base de connaissance liant les profils de numérisation aux différents types de documents et de déterminer si un document spécifique nécessite la création d'un nouveau profil de numérisation qui pourra être spécifié par l'utilisateur. Nous sommes donc typiquement dans le cadre d'un apprentissage incrémental à base de rejet avec lequel l'utilisateur peut interagir. Nous passons donc en revue ci-dessous, l'ensemble des concepts fondamentaux de l'apprentissage (R. Duda, 2001)[1] qui pourront servir pour l'élaboration du scanner cognitif.

Objectifs de l'apprentissage :

- **Classification** : Un problème de classification consiste à affecter des classes ou catégories à des individus décrits par des variables descriptives. On cherche donc à modéliser la relation entre des variables descriptives et une variable catégorielle représentant les classes. Ces classes peuvent être connues a priori (classification supervisée) ou inconnues (classification non supervisée). Dans le premier cas, l'apprentissage utilise une base d'individus exemples étiquetés – dont on connaît la classe - (apprentissage supervisé). Dans le second cas, l'apprentissage utilise une base d'individus non étiquetés (apprentissage non supervisé) pour définir lui-même les classes/catégories, en regroupant les individus semblables.
- Régression : Un problème de régression consiste à prédire la valeur d'une variable inconnue en fonction d'autres variables descriptives. En régression, on cherche donc à modéliser la relation entre cette variable inconnue et les variables d'entrées.

Modes d'apprentissage :

[1] Ces définitions sont essentiellement inspirées deBouillon, M. (2012). *Apprentissage incrémental et décrémental.* M2R Informatique, Insa de Rennes/IRISA.. Certaines correspondent à des notions bien établies dans la littérature. En Revanche d'autres correspondent à une vision plus personnelle.

- **Non supervisé, semi-supervisé, supervisé :** l'apprentissage supervisé exploite l'information catégorielle des données d'apprentissage (pour les problèmes de classification) ou les valeurs à prédire de celles-ci (pour la régression). L'apprentissage non supervisé n'exploite pas cette information (le plus souvent car elle est indisponible). L'apprentissage semi-supervisé (Zhu, 2005) permet d'exploiter à la fois des données étiquetées et non étiquetées. Il peut pour cela reposer sur différents modes d'apprentissage comme l'auto-apprentissage, l'apprentissage actif, le co-apprentissage (cf. ci-dessous), etc.
- **Statique/dynamique (adaptatif, incrémental, évolutif)** : dans l'apprentissage statique, le modèle (classification ou régression) est déterminé à partir des données d'apprentissage une fois pour toute. Dans un système dynamique, l'apprentissage se déroule tout au long de l'exploitation du système. On peut distinguer trois cas. Un système adaptatif sera capable de modifier son modèle au cours du temps pour s'adapter aux variations des données/du problème. Un système incrémental pourra construire son modèle petit à petit, au fur et à mesure que les données seront disponibles. Ces systèmes sont le plus souvent adaptatifs également par construction. Enfin, les systèmes évolutifs seront capables de changer leur modèle de façon plus importante pour par exemple résoudre un autre problème (nécessite une capacité d'oubli des connaissances).

Gestion des données :

- **Apprentissage en ligne/hors ligne** : que ce soit en apprentissage statique ou dynamique, les données d'apprentissage peuvent être utilisées soit toutes en même temps pour trouver le meilleur jeu de paramètres soit les unes à la suite des autres. Les méthodes d'optimisation globales utilisent plutôt l'apprentissage hors ligne, alors que les méthodes locales comme la descente de gradient utilisent plutôt un apprentissage en ligne. Par essence, l'apprentissage dynamique aura tendance à fonctionner par un apprentissage en ligne.
- **Mémoire des données** : lors d'un apprentissage dynamique, on conserve toutes les données sur lesquelles le système a été appris. A chaque réapprentissage, toutes les données et tout l'historique est donc disponible. C'est typiquement ce que ferait un k-nn incrémental. Bien entendu, la limite est le nombre de données à stocker et utiliser (problème de complexités). Une variante, consiste à ne conserver qu'une partie des données (mémoire partielle).
- **Mémoire des concepts** : dans ce mode, au lieu de conserver les données, on ne conserve que les propriétés des modèles appris par le système. Ainsi, lors d'un réapprentissage, on modifie directement le dernier modèle en fonction des nouvelles données, sans pour autant perdre ses propriétés précédentes (non régression). On est typiquement dans le cadre d'un système incrémental.

Méthodes spécifiques d'apprentissage :

- **Apprentissage incrémental** : l'apprentissage incrémental se réfère à deux situations qui ne sont pas forcément exclusives. La première est relative aux données qui ne sont pas toutes disponibles dès le début ou bien trop coûteuses à utiliser en une seule fois. Il s'agit donc de faire un apprentissage en ligne, au fur et à mesure de l'arrivée de nouvelles données pour élaborer un système qui

devienne de plus en plus performant. La deuxième situation, qui peut s'ajouter à de la précédente est que l'on ne connaît pas dès le départ toutes les propriétés du problème (ou que les données d'apprentissage ne les décrivent pas entièrement). Par exemple, ce peut–être le cas en classification supervisée ou non supervisée, lorsque l'on ne connaît pas a priori toutes les classes ni leur nombre exact. Dans ce cas, la structure même du modèle devient incrémentale. Si le système possède par ailleurs des capacités d'oubli (pour se focaliser sur le problème courant), on peut parler de système décrémental. Cet apprentissage peut donc aller d'une simple adaptation des paramètres du système (système adaptatif), à une évolution/transformation complète du modèle (système évolutif). Le problème principal lié à l'apprentissage incrémental est donc comment faire évoluer le modèle lorsque de nouvelles données sont disponibles. De ce problème en découle plusieurs autres. Concernant les données, est-ce que l'on effectue un apprentissage en ligne ou bien est-ce que l'on utilise une mémoire des données (complète/partielle) pour réapprendre ponctuellement tout ou partie du système ? Concernant le modèle, est-ce que l'on souhaite une mémoire des concepts (une non régression) ? Est-ce que l'on adapte essentiellement les paramètres ou la structure ? Dans ce dernier cas, comment gérer la création/suppression de classes, la modification d'une classe (paramètres), la fusion/division de classes ? Ces techniques ont été appliquées en premier lieu en classification non supervisée, notamment sur les systèmes à base de *clustering* notamment lorsque celles-ci utilisent des prototypes (K-means, etc.). En effet, dans celles-ci, la gestion incrémentale des données et la modification du modèle sont plus facile à faire. L'apprentissage incrémental a également été utilisé en classification supervisée, en particulier sur : les SVMs (Syed, 1999), (Ruping, 2001), (Boukharouba, 2009),(Masayuki Karasuyama, 2010) ; sur les ensembles de classifieurs qui sont élaborés de façon incrémental – ces méthodes sont dérivées du *bagging* et *boosting* - (Polikar, 2001), (Minku, 2009) ; sur les systèmes d'inférence floues, (Lughofer, 2008), (Kasabov, 2002), (Angelov, 2008), (Almaksour). La spécificité de ce dernier vient notamment de l'utilisation des rejets d'ambiguïté et de confusion(Mouchère, 2007), (Harold Mouchère, 2007) pour gérer l'aspect incrémental.

- **Apprentissage actif** (Settles, 2009), (Monteleoni, 2006) : il peut être vu comme une forme d'apprentissage semi-supervisé et peut également servir dans l'apprentissage dynamique. L'apprentissage actif s'utilise lorsque l'on dispose d'une grande quantité de données dont la plupart est peu étiquetées. Le principe consiste à sélectionner automatiquement une petite partie des données non étiquetées afin de les faire étiqueter par un oracle (souvent l'utilisateur). Ensuite, le système est réappris avec les nouvelles informations (après avoir éventuellement propager ces connaissances aux autres données non étiquetées. L'apprentissage peut ainsi être itéré. Cette méthode a été appliquée aux SVM et réseaux Bayésiens (Tong, 2001), pour la reconnaissance d'entités nommées(Olsson, 2008), etc. Il existe trois grands types de scénarios en apprentissage actif. On peut étiqueter potentiellement n'importe quel échantillon de l'espace et en particulier génèrer ceux qui sont les plus intéressants (*membership Query Synthesis*) (Angluin, 1988) Cette méthode présuppose que les données sont issues de distributions connues, voire uniformes et cela peut-être utile notamment en régression. Si les données ne sont pas issues de distributions uniformes, on peut prendre les données les unes à la suite des autres, de façon

séquentielle et au hasard, et l'algorithme décide de demander l'étiquette ou pas (*stream-based* ou *sequential active Learning*). Enfin, on peut sélectionner directement un sous-ensemble de données à étiqueter (*pool-based*). Dans ces deux derniers cas, l'algorithme peut reposer sur des mesures d'informativité.

- **Auto-apprentissage** (*self-training*) : utilisé notamment en apprentissage semi-supervisé, il repose sur un principe. Le système apprend un modèle à partir des données étiquetées puis utilise ce modèle pour étiqueter lui-même les données d'apprentissage non étiquetées avant de réapprendre entièrement son modèle avec l'ensemble des données d'apprentissage à présent étiqueté – on peut ne conserver que celle pour lesquelles le système est suffisamment sûr de l'étiquette qu'il a attribué (Rosenberg, 2005), (Culp, 2007), (Haffari, 2007). Cette méthode peut être vue comme parente d'un apprentissage actif dans lequel l'utilisateur qui étiquette (l'oracle) est le système lui-même.
- **Co-apprentissage** : le co-apprentissage repose sur un principe proche de l'auto-apprentissage mais cette fois, on utilise deux ou plusieurs algorithmes en parallèle. Les hypothèses initiales sont que l'on peut découper l'ensemble d'apprentissage en plusieurs parties conditionnellement indépendantes relativement à une classe et suffisante chacune pour pouvoir apprendre un classificateur(Blum, 1998). Chaque classifieur apprend sur ses données puis donne une étiquette sur les données non étiquetées de l'ensemble d'apprentissage pour lesquelles ils sont le plus confident. Chaque classifieur est alors réappris avec les nouvelles données de l'autre classifieur.
- **Apprentissage par renforcement** : Il s'agit là encore d'un apprentissage dynamique dans lequel le système cherche à optimiser une fonction en fonction de son expérience. A chaque fois qu'il prend une décision (relative à son modèle courant et à une nouvelle entrée/situation), il reçoit une récompense ou punition proportionnelle à la l'accroissement de la satisfaction de la fonction objectif.

Raisonnement à base de cas (Leake, 1996)[2]

Le raisonnement à base de cas n'est pas lié directement aux modes d'apprentissage qui viennent d'être présentés. Il s'agit plutôt d'un paradigme particulier dans lequel la classification peut s'inscrire (Instance Based Learning, (Cauchy, 2009)) et sur lequel un scanner cognitif pourrait s'appuyer. Le principe du raisonnement à base de cas consiste à s'appuyer sur les expériences passées (couples cas/situations et décisions prises) pour prendre les décisions courantes. Pour cela, la méthode repose sur 4 tâches :

- récupération de cas/situations issues des expériences passées et qui sont proches du cas/de la situation courante. Cette tâche repose sur l'utilisation de mesures de similarités ;
- Adaptation des décisions prises sur ces cas passés au cas courant ;
- Optimisation des solutions ainsi adaptées au problème courant ;
- Analyse et mémorisation de ces nouveaux cas dans la base de cas passés pour un usage futur.

[2] http://aaai.org/AITopics/CaseBasedReasoning ; http://www.aiai.ed.ac.uk/links/cbr.html ;

Bibliographie

Almaksour, A. *Incremental learning of evolving fuzzy inference systems : application to handwritten gesture recognition.* Institut National des Sciences Appliquées (INSA) de Rennes.

Angelov, P. a. (2008). Evolving fuzzy-rule-based classifiers from data streams. . *Fuzzy Systems, IEEE Transactions on , 16(6)*, 1462 –1475.

Angluin, D. (1988). Queries and concept learning. *Machine Learning , 2*, 319–342.

Blum, A. &. (1998). Combining labeled and unlabeled data with co-training. . *COLT: Proceedings of the Workshop on Computational Learning Theory.*

Bouillon, M. (2012). *Apprentissage incrémental et décrémental.* M2R Informatique, Insa de Rennes/IRISA.

Boukharouba, K. B. (2009). Incremental and decremental multi-category classification by support vector machines. *International Conference on Machine Learning and Applications, ICMLA '09*, (pp. 294 –300).

Culp, M. &. (2007). An iterative algorithm for extending learners to a semisupervised setting. . *The 2007 Joint Statistical Meetings (JSM) .*

Haffari, G. &. (2007). Analysis of semi-supervised learning with the Yarowsky algorithm. . *23rd Conference on Uncertainty in Artificial Intelligence (UAI) .*

Harold Mouchère, E. A. (2007). Writer Style Adaptation in On-line Handwriting Recognizers by a Fuzzy Mechanism Approach : The ADAPT Method. *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI) , 21(1)*, 99-116.

Kasabov, N. a. (2002). Denfis : Dynamic evolving neuralfuzzy inference system and its application for time-series prediction. *Fuzzy Systems, IEEE Transactions on , 10(2)*, 144 - 154.

Lughofer, E. (2008). Flexfis : A robust incremental learning approach for evolving takagi-sugeno fuzzy models. *Fuzzy Systems, IEEE Transactions on , 16(6)*, 1393 –1410.

Masayuki Karasuyama, I. T. (2010). Multiple incremental decremental learning of support vector machines. *IEEE Transactions on Neural Networks , 21/7*, 1048-1059.

Minku, F. L. (2009). Negative correlation in incremental learning. *Natural Computing : an international journal , 8*, 289–320.

Monteleoni, C. (2006). *Learning with Online Constraints: Shifting Concepts and Active Learning.* PhD thesis, Massachusetts Institute of Technology.

Mouchère, H. (2007). *Étude des mécanismes d'adaptation et de rejet pour l'optimisation de classifieurs : Application à la reconnaissance de l'écriture manuscrite en-ligne.* PhD Thesis, Institut National des Sciences Appliquées de Rennes (INSA).

Olsson, F. (2008). *Bootstrapping Named Entity Recognition by Means of Active Machine Learning.* PhD thesis, University of Gothenburg.

Polikar, R. U. (2001). Learn++ : An incremental learning algorithm for supervised neural networks. *IEEE Transactions on System, Man and Cybernetics (C), Special Issue on Knowledge Management , 31*, 497–508.

R. Duda, P. H. (2001). *Pattern Classification.* Wiley-Interscience.

Rosenberg, C. H. (2005). Semi-supervised self- training of object detection models. *Seventh IEEE Workshop on Applications of Computer Vision.*

Ruping, S. (2001). Incremental learning with support vector machines. *Proceedings IEEE International Conference on Data Mining, ICDM 2001*, (pp. 641 –642).

Settles, B. (2009). *Active Learning Literature Survey.* Computer Sciences Technical Report, University of Wisconsin–Madison.

Syed, N. A. (1999). Incremental learning with support vector machines. *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining KDD 99 .*

Tong, S. (2001). *Active Learning: Theory and Applications.* PhD thesis, Stanford University.

Zhu, X. (2005). *Semi-supervised learning literature survey.* Computer Sciences Technical Report, University of Wisconsin–Madison.