Google Panda : l'apprentissage automatique dans les algorithmes des moteurs de recherche

Retour au sommaire de la lettre

Domaine :	Recherche	**Référencement**
Niveau :	Pour tous	**Avancé**

La notion de "machine learning" ou "apprentissage automatique" est de plus en plus utilisée dans les algorithmes des moteurs de recherche et notamment dans le cadre de la lutte contre le spam. AInsi, plusieurs indices laissent à penser que Google Panda est un filtre basé sur ces techniques d'apprentissage automatique. Pour en savoir plus sur ce sujet, nous vous proposons un article qui a pour but de vous expliquer, de la façon la plus simple possible, les caractéristiques de ce type d'algorithme et leurs applications au quotiden par les moteurs de recherche. Mieux connaître, mieux comprendre, permet également de mieux gérer certaines situations...

Le "machine learning" (pour "apprentissage automatique", parfois appelé également "apprentissage artificiel") est un domaine de la science informatique qui s'est révélé extraordinairement prolifique depuis une quinzaine d'années. C'est en particulier l'une des disciplines de l'intelligence artificielle dont les applications pratiques se sont le plus répandu dans notre vie quotidienne. On retrouve des algorithmes d'apprentissage automatique dans de nombreux dispositifs de reconnaissance de forme, dans des logiciels d'assistance médicale, ainsi que dans la robotique …

Mais l'apprentissage automatique permet également de résoudre de manière élégante des problèmes complexes qui se posent aux moteurs de recherche. De nombreux travaux de recherche, datant parfois de plus de dix ans, ont exploré cette voie et démontré que le "machine learning" était une voie intéressante pour améliorer les algorithmes des moteurs, en particulier pour identifier les pages de web spam et améliorer de manière subtile la pertinence des résultats.

Nous verrons que quelques indices peuvent laisser penser que Google utilise déjà activement ces approches dans son algorithme, et que Panda, en particulier, présente des caractéristiques qui font penser à une approche de type "machine learning".

Qu'est-ce que l'apprentissage automatique ?

L'apprentissage automatique est une approche radicalement différente dans l'écriture de programmes informatiques permettant de résoudre des problèmes complexes.

Dans l'approche traditionnelle, l'objectif est prévoir et de décrire le type de données qui vont alimenter le programme et de programmer les traitements sur ces données. Cela signifie que l'on est en mesure de connaître à l'avance les caractéristiques des données en entrée, et les lois à appliquer à ces données pour obtenir le comportement souhaité du programme.

Mais dans de nombreuses circonstances, les données à traiter sont imprévisibles, trop complexes, ou difficiles à décrire. C'est le cas par exemple pour les logiciels de reconnaissance de forme ou de traitement d'image. Quant aux lois censées régir le traitement à apporter à ces données et produire les informations attendues en sortie, deux cas se présentent fréquemment :

- Les lois ne sont pas connues à l'avance, et les déterminer fait en réalité partie du problème à résoudre.

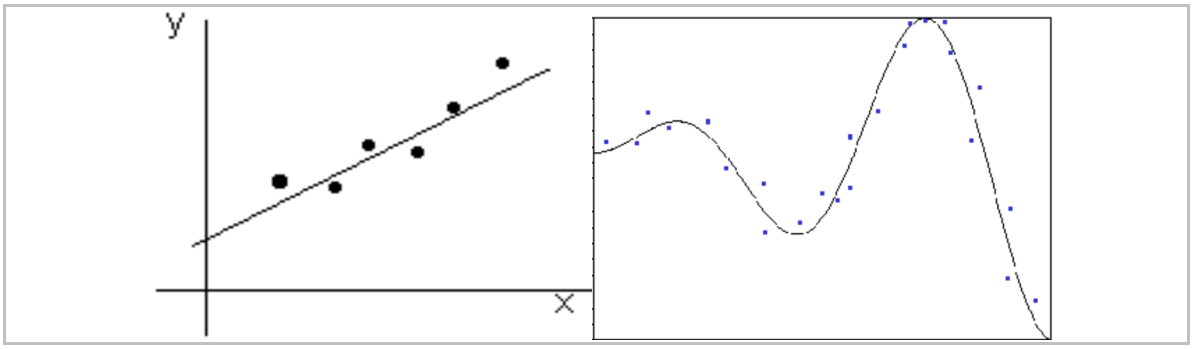
- Les lois seraient en théorie déterminables à l'avance, mais le nombre de cas différents à traiter et la complexité des lois rend leur programmation par des méthodes traditionnelles trop coûteuse. Dans certains cas, le nombre de paramètres à prendre en compte, la diversité des données en entrée et la complexité du comportement à produire en sortie est telle qu'il se produit une "explosion combinatoire" qui rend l'écriture d'un programme complètement irréaliste.

Pour résoudre ces problèmes, l'apprentissage automatique a été présenté comme une approche alternative intéressante.

L'idée de base derrière l'apprentissage automatique est :

- d'écrire des programmes qui peuvent apprendre et s'améliorer au fil du temps ;
- et/ou des programmes capables d'extraire des lois à partir d'échantillons de données, et de raisonner par induction (tirer des lois générales à partir d'un échantillon de données).

La définition de Tom Mitchell est souvent citée pour décrire le comportement des logiciels "apprenants" : *"On dit qu'un programme informatique apprend à partir de l'expérience E par rapport à la classe de tâches T et la mesure de performance P, si ses performances dans l'exécution des tâches de la classe T, mesurée par P, s'améliorent avec l'expérience E."*



Une première approche possible (et triviale) pour déterminer les lois régissant un ensemble de données : la régression. L'objectif est de déterminer une courbe qui donne une approximation correcte des points mesurés. Dans l'exemple de gauche ci-dessus, la courbe est droite affine y=ax+b. A droite, on trouve un polynome plus complexe.

Cette approche est devenue très populaire à partir des années 90 (dans le cadre d'un engouement général à l'époque pour l'intelligence artificielle). Depuis lors, elle a fait l'objet de nombreux travaux théoriques qui lui ont donné des fondements solides. Et l'apprentissage automatique a permis de réaliser des progrès spectaculaires dans de nombreux domaines, et ses applications pratiques sont nombreuses :

- Reconnaissance de forme, reconnaissance vocale et robotique ;
- Reconnaissance de caractères (OCR) ;
- Systèmes experts : logiciels de diagnostic médicaux, analyse financière, logiciels boursiers ;
- Fouille d'immenses bases de données : data mining ;
- Applications militaires, espionnage : détection de sous marin, analyse automatique de photos satellites.

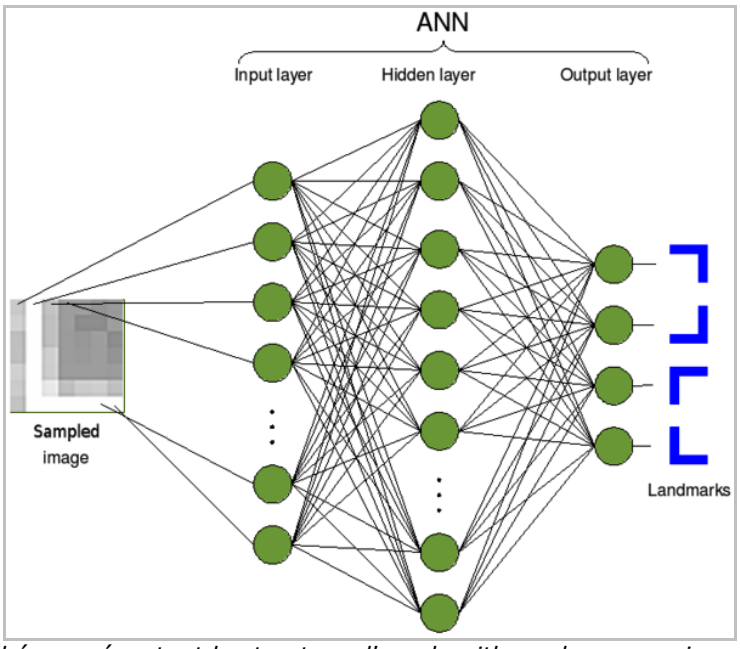


Schéma présentant la structure d'un algorithme de reconnaissance de forme utilisant la technique des "réseaux de neurones artificiel".

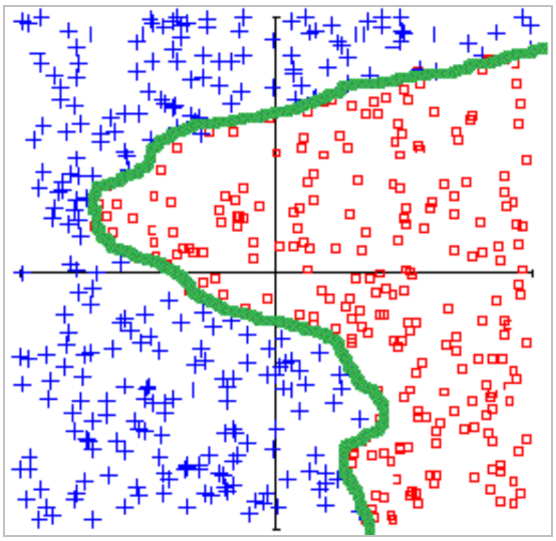
Parmi les applications pratiques, on trouve aussi les algorithmes antispam (en général, une application des algorithmes dits "bayesiens").

Les différents modes d'apprentissage

Les algorithmes d'apprentissage automatique sont souvent classés en fonction du type d'apprentissage qu'ils intègrent, et qui bien souvent détermine le type de comportement et de données en sortie que ces algorithmes peuvent produire.

On distingue :
- L'apprentissage supervisé ;
- L'apprentissage non supervisé ;
- L'apprentissage semi-supervisé ;
- L'apprentissage par renforcement ;
- L'apprentissage par transduction ;
- L'apprentissage multi-tâches ;

L'apprentissage supervisé s'appuie sur les choix faits par un "expert" baptisé Oracle, pour en tirer un modèle de classement. Généralement, la tâche de l'expert consiste à "labelliser" les données de l'échantillon, c'est-à-dire apposer une ou plusieurs "étiquettes".



Classification automatique utilisant l'apprentissage supervisé à partir de données étiquetées (classifiées) par un humain ("oracle"). L'objectif est de déterminer une fonction opérationnelle séparant correctement les points bleus et rouges (courbe verte)

L'apprentissage non supervisé correspond à des situations où l'on dispose d'exemples, mais pas d'étiquettes définies par un humain. Les algorithmes de clustering automatiques (algorithmes de "classification" ou de création de dendrogrammes) font généralement partie de cette classe.

L'apprentissage semi-supervisé qualifie des cas dans lesquels des étiquettes manquent sur les données d'entrainement en entrée. L'objectif dans ce cas est de déterminer néanmoins un comportement en sortie, en raisonnant soit de manière probabiliste, soit en termes de ratio coût/bénéfice (analyse de risque).

L'apprentissage par renforcement décrit un algorithme qui cherche à maximiser les gains sur le long terme en analysant les séquences : perception => décision/action => résultats. Si les résultats sont évalués comme constituant un gain, ce "modèle de décisions" doit être favorisé. Ce type d'algorithme est souvent utilisé en robotique.

L'apprentissage par transduction peut être utilisé quand raisonner du particulier au général (induction) ne résout pas le problème. Raisonner par transduction consiste à procéder par analogies. Même s'il est connu que ce mode de raisonnement produit des syllogismes, il peut produire des résultats intéressants dans des cas bien balisés où la recherche d'une loi générale est trop coûteuse en temps machine ou franchement sans intérêt.

L'apprentissage multi-tâches (Learning to learn) est une approche inductive, dans lequel on cherche à trouver un modèle adapté à plusieurs tâches à la fois. Le programme apprend des expériences sur chaque classe de tâches, mais en même temps cherche à réutiliser ce qu'il a appris d'efficace pour une tâche A sur une tâche B.

Les algorithmes les plus utilisés

Il existe un grand nombre d'algorithmes d'apprentissage automatique, nous ne citerons que les plus courants, en précisant qu'il existe de très nombreuses variantes au sein de chaque "famille" d'algorithmes :
- Les algorithmes à bases d'arbres de décision ;
- L'apprentissage de règles d'association ;
- Les réseaux de neurones artificiels ;
- Les algorithmes génétiques ;
- L'ILP (inductive logic programming) : Programmation logique inductive ;
- Les Support Vector Machines.
- Les Réseaux Bayésiens.

Expliquer les caractéristiques de chacun de ces algorithmes dépasse de loin la portée de cet article, nous renvoyons donc les lecteurs curieux à la bibliographie placée à la fin de l'article où deux livres d'initiation sont cités.

L'apprentissage automatique dans les moteurs de recherche - la lutte contre le spam

Les premiers travaux envisageant l'utilisation d'algorithmes d'apprentissage automatique pour créer un algorithme de classement remontent à une quinzaine d'années. Mais les premières démonstrations d'applications pratiques efficaces pour ces algorithmes datent de 1999/2000 et concernent un cas particulier, pour lequel ces algorithmes sont particulièrement adaptés, à savoir la lutte contre le spam.

Pourquoi l'apprentissage automatique est-il adapté à la lutte contre le spam ?

En effet, si l'on veut lister les critères qui font qu'un évaluateur humain va classer (étiqueter) une page comme étant du spam, on arrive vite à une liste assez longue. Ensuite, on s'aperçoit facilement que cet évaluateur humain prend des décisions complexes, rarement binaires : c'est la présence combinée de plusieurs facteurs qu'il prend en compte, en tenant compte en plus de l'importance de chaque facteur l'un par rapport à l'autre. Et même l'observation des facteurs pris en compte, et des poids de ces facteurs ne suffit pas à "prédire" correctement la décision prise par les humains : les évaluateurs décident en fonction de "patterns", de motifs qu'ils reconnaissent spontanément.

Les approches consistant à programmer une fonction d'évaluation du caractère "spammy" ou non d'une page se sont longtemps heurtées à ces subtilités. En fonction du caractère plus ou moins stricts des réglages, elles généraient soient beaucoup de "faux positifs", soit beaucoup de cas "gris", soit même les deux à la fois. Le recours à des évaluateurs humains pour éliminer le spam reste donc obligatoire, ces fonctions d'évaluation servant plus sûrement à étiqueter les cas "simples" faciles à discriminer, le reste étant soumis à une évaluation humaine.

Le problème c'est que l'ampleur de la tâche est telle que ce système est condamné à l'avance : le webspam se développe sans arrêt, prend des formes de plus en plus complexes, le nombre de requêtes tapées par les internautes explose et le nombre de pages indexées a cru considérablement ces dernières années. Utiliser des humains pour éliminer le spam représente un coût beaucoup trop élevé, et n'est pas une solution facile à étendre à toutes les langues, tous les pays en conservant un niveau de qualité uniforme.

Il est donc assez naturel de penser aux algorithmes d'apprentissage automatique pour faire face à cette complexité et limiter le coût en temps / homme nécessaire pour traiter ce problème.

Pourquoi ces algorithmes ont-ils été longtemps sous-utilisés par les moteurs de recherche ?

Longtemps, l'utilisation de ces algorithmes a été bridée par trois problèmes :

- **La difficulté de création des données d'entrainement** : ces algorithmes sont efficaces à condition que l'échantillon de données en entrée soit suffisamment important pour couvrir tous les cas à traiter. Or, s'agissant du web, le volume de données nécessaire pour créer un échantillon correctement étiqueté est particulièrement important. Le travail d'étiquetage demande donc aussi un travail important de la part d'"experts humains". Le ratio coût/bénéfices du recours à l'apprentissage automatique n'a donc pas été jugé toujours favorable.

- **Le temps de calcul et les besoins en ressources** : plus l'échantillon à analyser est gros, plus le calcul demande du temps (qui peut se compter en… semaines) et réclame des ressources importantes. Or, au-delà d'un certain seuil, cela n'a plus de sens, car si la mise au point d'un algorithme de détection du spam ne peut pas suivre le rythme d'évolution du web, utiliser cette approche n'a pas de sens.

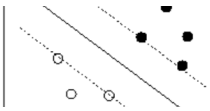
- **La faisabilité** : pour résoudre certains cas, cette approche peut se révéler d'emblée totalement irréaliste. Au-delà d'une certaine complexité, on peut démontrer que ces algorithmes sont souvent incapables de fournir des résultats dans un temps "polynomial", c'est-à-dire borné par un polynôme de la taille des données.

Les progrès effectués, notamment sur les SVM (Support Vector Machines) étendent le champ d'application de l'apprentissage automatique.

Des progrès rapides ont néanmoins été accomplis ces dernières années, à la fois sur le plan des bases théoriques, des mathématiques sous-jacentes, et de l'implémentation de ces algorithmes.

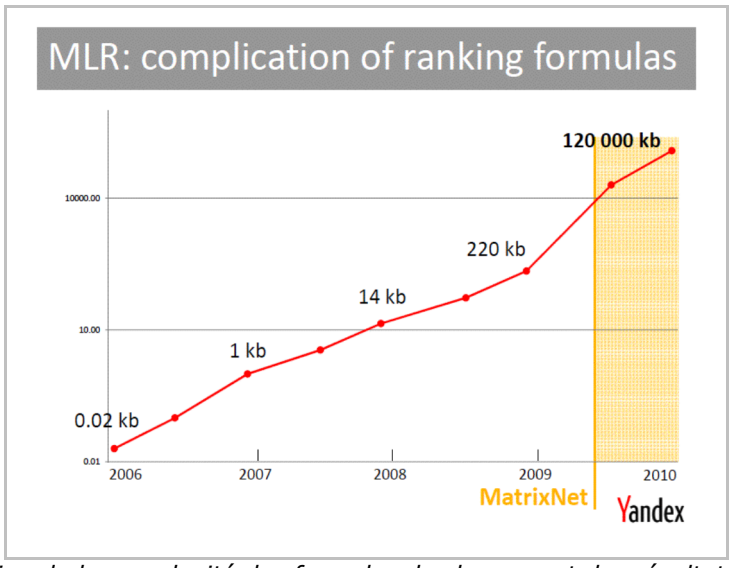
L'approche la plus fructueuse pour une application dans les moteurs de recherche est probablement représentée aujourd'hui par les SVM : les "*Support Vector Machines*". En français, on appelle ces techniques d'apprentissage automatiques soit Machines à Vecteurs Supports, soit Séparateurs à Vastes Marges (ce qui décrit bien leur principe tout en préservant le même acronyme qu'en anglais).

Les SVM sont nés des travaux théoriques de Vladimir Vlapnik. Cette approche s'est fortement développée à la fin des années 90, pour devenir très populaire aujourd'hui compte tenu de ses nombreux avantages : les SVM permettent de traiter de gros volumes de données avec une efficacité supérieure à de nombreuses approches, minimisent les erreurs par construction, et s'appuient sur des algorithmes que l'on sait optimiser.



Un des principes de la technique des SVM : il existe souvent plusieurs frontières permettant de "séparer" des échantillons de points. Les meilleures frontières sont celles qui maximisent la distance entre la frontière et les points (les "séparateurs à vastes marges"). L'algorithme SVM permet d'identifier les frontières donnant le minimum d'erreurs de classification.

Il n'est donc pas surprenant de constater que la plupart des grands moteurs de recherche (Yahoo!, Google, Bing…) ont publié ces dernières années de nombreux brevets et publications sur l'utilisation potentielle de l'apprentissage automatique en général, et des SVM en particulier dans différents domaines.



Courbe d'évolution de la complexité des formules de classement des résultats dans le moteur Yandex, leader en Russie. L'utilisation des algorithmes d'apprentissage automatique (MLR) a fait passer en un an la taille des formules de 220 000 octets à 120 méga octets. L'échelle est logarithmique : la croissance est en réalité exponentielle.

Le machine learning et Google

Les chercheurs de Google ont publié, comme leurs homologues chez Bing et Yahoo !, un nombre de publications impressionnant sur l'apprentissage automatique. On trouve une plus grande concentration de travaux sur les SVM chez Google toutefois. Ceci peut peut-être s'expliquer par le fait qu'une certaine Corrina Cortes est à la tête du centre de recherche de Google à New York. Corrina Cortes est l'une des spécialistes mondialement reconnue des SVM, et a notamment contribué avec Vladimir Vlapnik à en poser les bases théoriques.

Mais au-delà de cette activité de recherche prolifique, on trouve trois références en particulier qui laissent penser que la firme basée à Mountain View s'intéresse de près au "machine learning", au point de se doter récemment d'une infrastructure adaptée pour utiliser ces algorithmes à grande échelle.

Tout d'abord, une publication datant de 2009 décrit une architecture, baptisée **PLANET** capable de gérer des algorithmes de type "arbres de décisions" en utilisant une architecture massivement parallèle.

Ensuite, plusieurs personnes de Google ont présenté au colloque LADIS 2010, un système baptisé **SYBIL**, facilitant les calculs complexes nécessaires pour les algorithmes d'apprentissage automatique, utilisable notamment pour la détection du spam, et s'appuyant sur l'infrastructure habituelle de Google notamment MapReduce et GFS.

Panda s'appuie-t'-l sur un algorithme d'apprentissage automatique ?

Plusieurs indices laissent penser que Panda est probablement une application d'un algorithme d'apprentissage automatique.

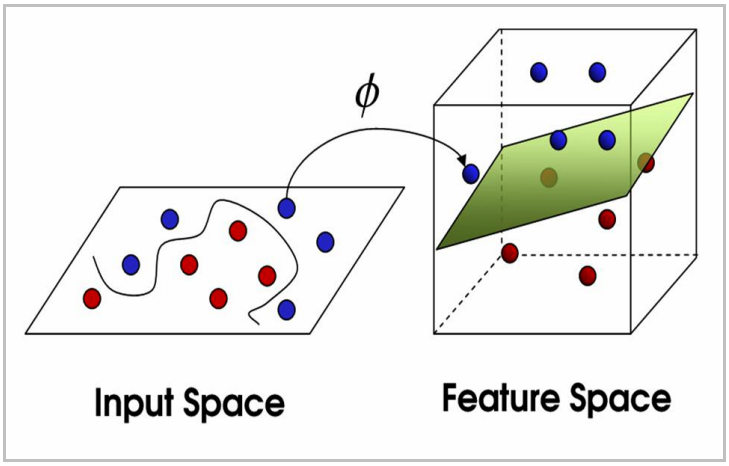
Tout d'abord, le père de l'algorithme est un certain Panda. Or il existe deux ingénieurs travaillant chez Google et qui portent ce patronyme : Navneet Panda et Biswanath Panda. Il s'avère que ces deux Panda ont publié des publications dans le domaine du "Machine Learning".

Dans le même temps, la façon dont Matt Cutts et Amit Singhal ont décrit l'algorithme rappelle quelques caractéristiques de ces algorithmes. Tout d'abord, on sait que :

- L'algorithme prend en compte les données d'un échantillon de données fourni par des évaluateurs humains ;
- Depuis le 11 avril, on sait qu'il prend en compte également les données fournies par l'extension du navigateur Chrome "personal blocklist".

Amit Singhal décrit l'algorithme ainsi dans une interview de Wired: "*Vous pouvez imaginer dans un espace muldimensionnel un groupe de points, certains points sont rouges, certains points sont verts, and pour d'autres c'est un mélange des deux. Votre travail est de trouver un hyperplan qui indique que la plupart des choses d'un côté de ce plan sont rouges, et que la plupart des choses de l'autre côté sont le contraire de "rouge"*".

Cette description rappelle fortement, pour les initiés, la recherche d'une fonction "noyau" (kernel) dans la technique des SVM !



Recherche d'un hyperplan "frontière" dans la technique des SVM : la frontière ici est relativement complexe si on la décrit dans l'espace à deux dimensions qui décrit les données d'entrainement. En transposant le problème dans un espace muldimensionnel (3 dimensions sur le schéma) il peut être possible de trouver un hyperplan (ici un plan), simple à décrire, qui permet de classifier facilement les données.
Ce schéma correspond de manière troublante à la description d'Amit Singhal

D'autres indices, plus légers ceux-là, confirment qu'un algorithme de ce type a servi pour Panda : l'algorithme semble être universel, mais nécessite des calculs spécifiques à partir des données locales avant d'être déployé. Le temps de calcul semble être long, puisque la mise à jour de l'algorithme ne semble pas pouvoir se faire régulièrement…

Qu'en déduire quant au fonctionnement de Panda ?

Si Panda est un algorithme d'apprentissage automatique, le problème du reverse engineering de l'algorithme peut très bien se révéler parfaitement insoluble… D'abord parce que ce type d'algorithmes permet d'intégrer de nombreux paramètres dans une décision : savoir quel critère a fait basculer votre site dans l'infamie devient tout de suite plus difficile. Ensuite, ces algorithmes sont connus pour rendre (souvent) opaques le processus qui conduit le programme à classer une page dans une catégorie donnée, y compris pour les personnes qui ont créé le système d'apprentissage. La rétroingéniérie du système est dans ces conditions assez vaine.

Donc connaître la nature de l'algorithme ne permet pas de savoir "comment sortir de Panda". Seule la détermination des critères "humains" de classement dans la liste des sites à éliminer peut contribuer à voir ce qu'il faut faire. Mais la liste des critères publiés par Amit Singhal semble être incomplète car elle porte avant tout sur le cas des "content farms" et pas sur les autres sites touchés par le nouvel algorithme…

L'apprentissage automatique : l'avenir de la lutte contre le web spam ?

S'il se confirme que Panda s'appuie bien sur un tel algorithme, on assiste donc à un véritable tournant. Le recours par Google à un tel algorithme pour résoudre un problème complexe de Web Spam, signifie deux choses :
- Les problèmes de scalabilité ont été résolus de manière satisfaisante ;
- Et les résultats sont suffisamment bons pour que l'on puisse se passer d'une intervention humaine massive.

Dans la lutte éternelle entre l'épée (les spammeurs) et la cuirasse, il semble qu'après des années où la cuirasse ne couvrait pas suffisamment les parties vulnérables du moteur, et comportait de nombreux défauts, une avancée significative permet aux moteurs de reprendre (partiellement) l'avantage. D'autres progrès sont-ils à attendre grâce à l'apprentissage automatique et l'intelligence artificielle ? Probablement. Cela suffira-t-il à donner un avantage sérieux et durable aux moteurs contre les spammeurs : l'avenir seul nous le dira…

Bibliographie

Livres pour approndir le sujet :

Apprentissage artificiel, Concepts et algorithmes
Auteur(s) : Antoine Cornuéjols , Laurent Miclet, Editeur : Eyrolles

Apprentissage statistique, Réseaux de neurones - Cartes topologiques - Machines à vecteurs supports
par Gérard Dreyfus , Jean-Marc Martinez , Manuel Samuelides , Mirta B. Gordon , Fouad Badran , Sylvie Thiria chez Eyrolles .

A propos de l'update Florida et des algorithmes bayésiens :

Quelques pistes pour comprendre le nouvel algorithme de Google
http://www.webmaster-hub.com/publication/Quelques-pistes-pour-comprendre-le.html
Philippe Yonnet, Juillet 2004

A propos de l'état de l'art du machine learning dans les moteurs de recherche :

Learning with support vector machines and rational kernels
Conférence de Corinna Cortes à l'occasion du colloque Maths Avenir à Paris, 2009 :
http://www.maths-a-venir.org/2009/expos%C3%A9-corinna-cortes

Publications scientifiques

Optimizing Search Engines using Clickthrough Data
Thorsten Joachims, Cornell University
http://www.cs.cornell.edu/people/tj/publications/joachims_02c.pdf

Detecting Spam Blogs: A Machine Learning Approach
Pranam Kolari, Akshay Java, Tim Finin, Tim Oates, Anupam Joshi
University of Maryland Baltimore County
http://aisl.umbc.edu/resources/260.pdf

Machine Learning chez Google :

Publications by Googlers in Machine Learning
http://research.google.com/pubs/MachineLearning.html

PLANET: Massively Parallel Learning of Tree Ensembles with MapReduce
Biswanath Panda, Joshua S. Herbach, Sugato Basu, Roberto J. Bayardo, Google, Inc.
http://www.bayardo.org/ps/vldb2009.pdf

KDX: An Indexer for Support Vector Machines
Navneet Panda, Edward Y. Chang, Google Inc
http://www.computer.org/portal/web/csdl/doi/10.1109/TKDE.2006.101

OASIS : Large Scale Online Learning of Image Similarity Through Ranking
Gal Chechik , Varun Sharma, Samy Bengio, Google Inc & Uri Shalit, The Gonda brain research center, Bar Ilan University
http://www.robots.ox.ac.uk/~vgg/rg/papers/rankingsimilarity.pdf

Machine Learning chez Bing!

http://research.microsoft.com/en-us/groups/ml/
http://research.microsoft.com/en-us/groups/mlp/
http://research.microsoft.com/en-us/groups/mlpml/
...

Beyond PageRank: Machine Learning for Static Ranking
Matthew Richardson, Microsoft Research, Amit Prakash MSN, Eric Brill, Microsoft Research
http://www.inf.unibz.it/~ricci/SDB/slides/fRank-Presentation.pdf

Machine Learning chez Yahoo!

http://research.yahoo.com/Machine_Learning

Developing parallel sequential minimal optimization for fast training support vector machine.
Yahoo Labs, Cao, L.J.; Keerthi, S.S.; Ong, C.J.; Uvaraj, P.; Fu, X.J.; Lee, H.P.
http://research.yahoo.com/pub/951

***Philippe YONNET** , Directeur SEO international, Twenga.*

Réagissez à cet article sur le blog des abonnés d'Abondance :
http://blog-abonnes.abondance.com/2011/06/google-panda-lapprentissage-automatique.html