Penguin 3 : comment prévoir la vulnérabilité de votre site ?

Par Guillaume et Sylvain Peyronnet

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

La troisième mouture du filtre Penguin, luttant contre les liens de faible qualité, a été lancée en octobre 2014. Au travers de cet article, vous découvrirez comment, en utilisant des outils d'analyse avancée (classifieurs, algorithmes et arbres de décisions), il est possible de construire un système prévisionnel permettant de savoir si votre site est potentiellement à l'abri de ce type de pénalité. Ou pas...

Penguin est, vous le savez sans doute déjà, le nom d'un algorithme qui a pour but de pénaliser les pages qui ont des pratiques contraires à celles compatibles avec les *guidelines* de Google. Contrairement à l'algorithme Panda, Penguin n'utilise pas comme référence des jugements humains (le quality rating) mais des critères techniques et à pour but de minimiser l'impact des techniques de référencement les plus agressives (ce que l'on appelle généralement le SEO Black Hat).

Dans cet article, nous nous intéresserons à la troisième mise à jour de l'index de Google par cet algorithme. Cette mise à jour a eu lieu le 17 octobre 2014, et vise essentiellement les abus concernant les liens (voir [1]). Cet événement a été très commenté, et naturellement, beaucoup de questions ont été soulevées dans les semaines qui ont suivi (et encore maintenant). Ainsi, de nombreux webmasters se demandent si ils ont été touchés, et pourquoi ils l'ont été, de manière à faire les corrections nécessaires. Cette information sur le « pourquoi » est également utile pour ceux qui sont passés à travers le filtre, mais qui craignent que cela ne soit pas le cas la prochaine fois…

La différence entre aujourd'hui et il y a quelques années est que la communauté SEO a su monter en compétences et se doter de nouveaux services utiles. Ainsi, il existe actuellement de nombreux outils de

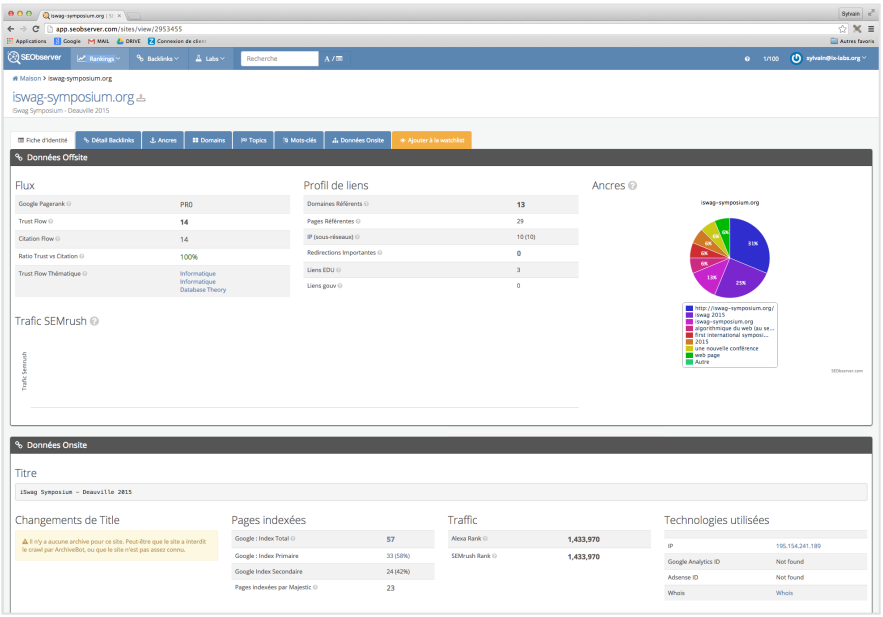

Fig.1. Copie d 'écran de SEObserver

suivi de positionnement, analyse de log, analyse sémantique, etc. Parmi ceux-ci, on compte un outil qui s'appelle SEObserver, développé par Kevin Richard (voir [2,3]. L'outil fournit de nombreuses informations : trustflow, citationflow, positionnement, backlinks et ancres associées, etc.

A l'aide de cet outil, on peut agréger des données qui permettent de comprendre ce qui se passe lors de changements majeurs dans les SERPs, comme par exemple lors du déploiement d'un filtre tel que Panda ou Penguin. En pratique, ces données sont utiles et suffisantes pour analyser la mise à jour Penguin, mais pas Panda, car pour cette dernière des critères non techniques sont utilisés, qui ne sont pas captés par un outil d'analyse tel que Seobserver.

Voyons maintenant ce que l'outil peut nous montrer, ce que les référenceurs en font, ainsi que son utilisation grâce à des outils de classification.

Les données et l'avis du référenceur

Sur le blog de Seobserver [4], Kevin Richard fournit des données bruts issues de l'outil. Il s'agit d'une captation, sur un échantillon représentatif de 3 000 mots-clés, des gains et pertes de positions sur google.fr lors du déploiement de Penguin 3.

Les 3 000 mots-clés choisis sont des termes hors marque pris parmi les plus importants pour Adwords (en utilisant un score qui est le produit du volume de recherche, du CPC et du niveau de concurrence). On a donc surtout à notre disposition dans les données des mots-clés compétitifs, ce qui n'est pas nécessairement un problème puisque les sites présents pour ces requêtes sont sans aucun doute les plus susceptible d'utiliser des techniques de référencement plus agressives et donc les plus enclins à être pénalisés.

Ensuite, les données ne s'intéressent qu'aux sites ayant eu un mouvement de cinq places au moins (en plus ou en moins), et se trouvant dans les 30 premières positions pour Google.fr avant le passage du filtre. Nous avons été encore plus restrictif de notre coté, puisque nous n'avons regardé que les sites qui perdaient au moins 10 positions. Là encore, les hypothèses sont raisonnables : on veut éliminer les petites variations parasites, et on ne s'intéresse qu'aux sites qui ont la capacité à se positionner.

Concernant les caractéristiques des pages, voici celles qui sont les plus importantes parmi celles disponibles :

- Volume du mot-clé selon Google Adwords ;
- PageRank. Ici il s'agit bien de ce que l'on appelle également la « barre verte » ;
- Nombre de domaines référents ;
- Citation Flow. Il s'agit de la métrique de popularité proposée par Majestic SEO ;
- Trust Flow. La métrique de confiance de Majestic SEO ;
- Le nombre de domaines référents avec liens ayant une ancre qui est un mot-clé exact ;
- Le nombre de domaines référents avec liens ayant une ancre qui contient un mot-clé (par exemple ancre « voiture cabriolet super rapide » pour le mot-clé « cabriolet ») ;
- Les nombres de pages référentes avec ancres exactes ou plus larges (même notion que pour les domaines référents, mais pour les pages uniquement) ;
- Cost-Per-Click du mot-clé. Indicateur de compétitivité du mot-clé ;

• Niveau de concurrence du mot-clé ;
• Position avant et après le passage du filtre Penguin ;
• Alexa rank.

Parmi tous ces critères, on trouve donc des indicateurs classiques (PageRank, Alexa rank, Trust Flow et Citation Flow), mais également les critères qui vous nous intéresser particulièrement : les liens très optimisés.

Avec ces données, Kevin Richard a pu écrire un article dans le JDN (voir [5]) où il explique ce qui est utilisé par le filtre pour prendre une décision de pénalisation. Selon lui, le critère qui est utilisé est celui du nombre d'ancres exactes. Il dit même ceci : *«… visuellement, le graphique de répartition des ancres nous a sauté aux yeux. On constate en effet une très grosse concentration d'ancres exactes optimisées, au minimum 10 % voire beaucoup plus ! ».*

La question que nous nous posons dans cet article n'est pas de savoir si cette conclusion est correcte (elle l'est), mais de savoir si on peut trouver de manière automatique les critères utilisés par un filtre comme Penguin, et si on peut créer un outil de prédiction de ce qu'il convient d'appeler la « penguinisation » ?

A ces deux questions la réponse est positive, et nous allons voir dans le reste de cet article comment cela est possible.

Classification et prédiction

La classification est une sous-discipline de l'apprentissage automatique (*Machine Learning* en anglais), qui a pour objectif de placer des individus dans des classes, c'est-à-dire des ensembles qui partagent des caractères statistiques. Ici, on va utiliser une méthode de classification, pour comprendre quels sont les caractères

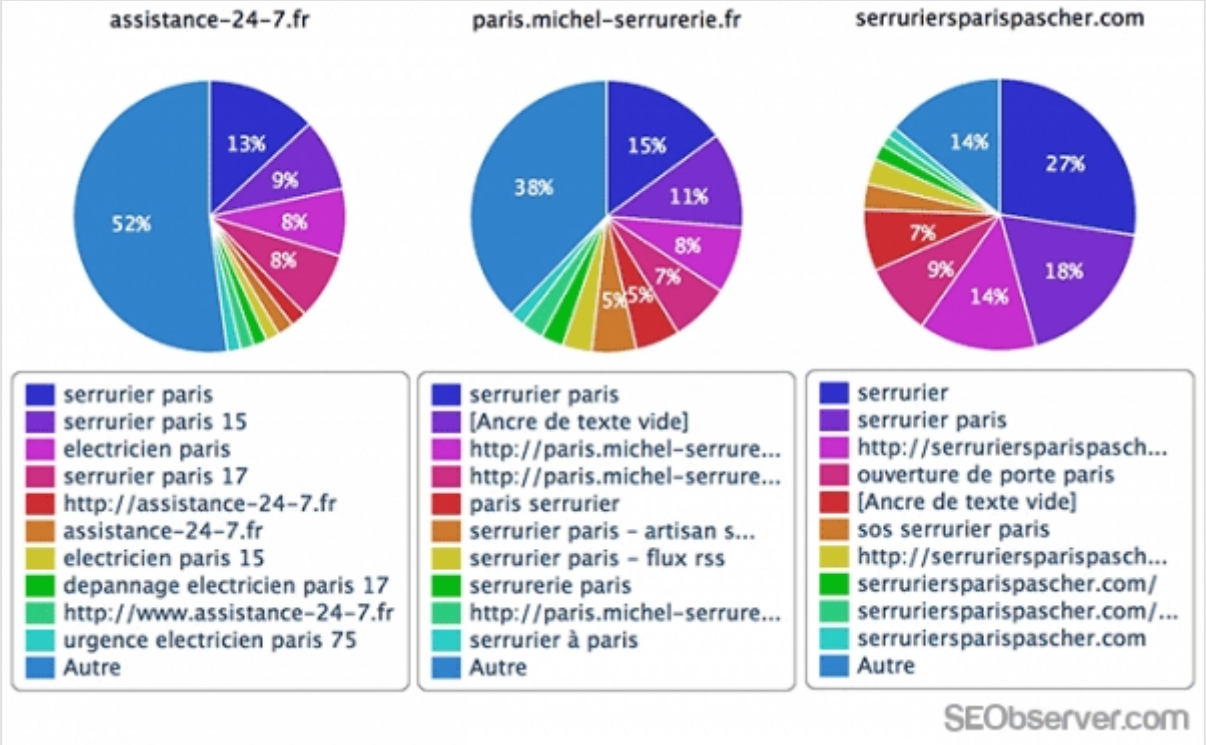

Fig.2. Copie d'écran montrant les ancres exactes pour quelques sites, tiré de [5]

statistiques de « la classe des pages web qui ont été très pénalisés par Penguin 3 ». Une fois cette compréhension acquise, on va mettre en place un mécanisme de prédiction, de manière très simple : si on repère un motif de classification qui est caractéristique des pages pénalisés, alors on prédira que toutes les pages qui présentent ce même motif courent le risque d'être pénalisées également.

La méthode de classification que nous allons utiliser permet de générer ce que l'on appelle un arbre de décision. Il s'agit d'un ensemble de « questions » utilisées dans un ordre spécifique, et qui va permettre de prendre une décision concernant une classification.

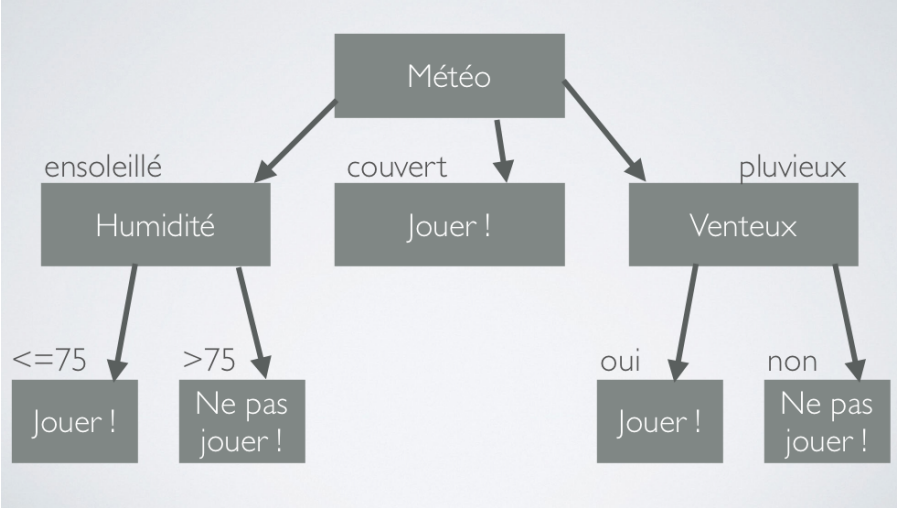

Fig.3. Exemple d'arbre de décision

La figure 3 montre l'exemple canonique d'arbre de décision présenté par Ross Quinlan dans son livre [6]. Il permet de modéliser le comportement d'un sportif qui se demande avant de sortir pour s'entrainer si il va pleuvoir ou non. Pour cela, le sportif regarde l'état de la météo : si elle est couverte, il sort car il ne pleuvra pas ; si le temps est ensoleillé, il regarde le niveau d'humidité, si ce dernier est supérieur à 75%, alors il va sans doute pleuvoir ; etc.

L'utilisation d'un arbre de décision est très intuitive, et permet de déterminer des règles très précises telles que « si il y a du soleil et pas d'humidité, il ne pleuvra pas ». Pour construire un arbre de ce type, il faut utiliser un algorithme de classification sur un ensemble de données bien choisies (ce que l'on appelle un dataset). Par exemple, pour l'arbre de la figure, il faut observer la météo pendant des semaines pour récolter suffisamment de données. L'algorithme que nous allons utiliser par la suite est le C5.0, son implémentation est libre et disponible sur le site de la référence [7].

L'algorithme C5.0 a été mis au point par Ross Quinlan, chercheur en fouille de données et intelligence artificielle. Il a tout d'abord inventé l'algorithme ID3, qui a ensuite évolué vers le C4.5 puis le C5.0. Cette série d'algorithmes fait partie du top 10 des algorithmes de fouille de données (voir [9]). Le C5.0 est très précisément un algorithme de classification supervisé, qui produit des classifieurs via un procédé de maximisation du gain informationnel. Nous n'expliquerons pas ici la théorie (complexe) qui se cache derrière l'algorithme.

Autant la théorie est compliquée, autant la pratique est simple, puisqu'il suffit de fournir au logiciel C5.0 les données (bien choisies) et les caractéristiques à étudier, ainsi que le nom des classes, et tout le reste se fait de manière (presque) automatique. Le « presque » étant là pour

dire que si l'on souhaite un excellent résultat, il faut modifier le paramétrage du logiciel à la main.

Nous allons maintenant voir ce que l'on peut faire avec cet algorithme C5.0 sur les données issues de l'outil seobserver lors du déploiement du penguin 3.

Utilisation du C5.0 dans le contexte du Penguin 3

Notre objectif ici est donc de créer un classifieur le plus efficace possible. Pour cela, nous avons réalisé une forêt d'arbres de décision avec une méthode de *boosting*. Un arbre de décision à toujours un certain pourcentage d'erreur. L'idée derrière la forêt d'arbres de décision est de dire que les erreurs d'un arbre ne sont pas nécessairement celles d'un autre, et que donc on peut améliorer la prédiction/classification si on a plusieurs arbres en parallèle.

Nous avons tout d'abord constitué notre dataset en prenant 4001 pages observées avec l'outil. Pourquoi 4001 ? Il n'y a pas d'autre raison que le fait que ce dataset a fourni de bons résultats. Nous avons conservé tous les critères mentionnés en introduction de cet article, en filtrant cependant les résultats pour ne garder que des pages qui ont subi de très fortes décotes : perte d'au minimum 10 places dans les SERP.

Ensuite, nous avons créé un fichier *seobserver.names*, qui contient une description des critères et des classes. La figure 4 montre le contenu de ce fichier.

Les deux classes sont notées *pe* (pour « penguin ») et *npe* (pour « no penguin »). Les autres lignes indiquent les critères, et le fait qu'ils soient codés par des valeurs numériques (« continuous »). Un deuxième fichier (*seobserver.data*) contient les données, en format proche du format CSV. Nous avons ensuite lancés le C5.0 sur les données en question, en tapant « c5.0 –f seobserver –b » en ligne de commande.

```
penguin.

KeywordVolume:                  continuous.
KeywordCpc:                     continuous.
KeywordCompetition:             continuous.
Pr:                             continuous.
Alexa:                          continuous.
RefPages:                       continuous.
RefDomains:                     continuous.
CitationFlow:                   continuous.
TrustFlow:                      continuous.
YesterdayPos:                   continuous.
ExactAnchorsRefDomains:         continuous.
ExactAnchorsRefPages:           continuous.
ContainingAnchorsRefDomains:    continuous.
ContainingAnchorsRefPages:      continuous.

penguin: pe, npe.
```

Fig.4. Contenu du fichier seobseserver.names

On obtient alors une forêt de classification avec les résultats de la figure 5.

Cette forêt d'arbre de décisions permet d'obtenir un classifieur avec une probabilité d'erreur totale de 5,5%. Si on regarde plus précisément cette erreur, elle est largement asymétrique : 2% de faux positifs et 28% de faux négatifs. De nombreux sites « spammy » sont donc passés entre les mailles de notre filet.

Un point intéressant est de voir les critères les plus utilisés par le classifieur obtenu (figure 6).

On voit que les premiers critères sont les positionnements avant le filtrage, la compétitivité du mot-clé et les citation

flow, trust flow et PR. Cela signifie que les sites lourdement pénalisés sont parmi ceux qui ont des bons indicateurs de qualité et qui sont dans des domaines concurrentiels. Ce n'est pas étonnant : le moteur n'a pas d'intérêt à pénaliser un site « spammy » qui ne se positionne pas bien.

```
Evaluation on training data (4001 cases):

Trial        Decision Tree
-----      ----------------
           Size      Errors

   0         18   347( 8.7%)
   1          7   574(14.3%)
   2         16   403(10.1%)
   3         20   437(10.9%)
   4         24   393( 9.8%)
   5         24   453(11.3%)
   6          4   477(11.9%)
   7         23   563(14.1%)
   8         11   419(10.5%)
   9         25   480(12.0%)
boost            220( 5.5%)   <<
```

Fig.5. Forêt de classification obtenue avec l'algorithme c5.0

On voit ensuite apparaître le critère du

```
  Attribute usage:

       90%  YesterdayPos
       40%  KeywordVolume
       40%  KeywordCpc
       40%  KeywordCompetition
       40%  CitationFlow
       40%  TrustFlow
       40%  Pr
       39%  ExactAnchorsRefDomains
       35%  Alexa
       35%  RefDomains
       34%  RefPages
       33%  ContainingAnchorsRefDomains
       23%  ExactAnchorsRefPages
       21%  ContainingAnchorsRefPages
```

Fig.6. Critères les plus utilisés par le classifieur

nombre de domaines référents avec ancres exactes, puis un peu plus loin le nombre de domaines référents et de pages référentes. On a donc une confirmation de ce qui était annoncé par les observateurs : avoir un grand nombre de mots-clés avec ancres exactes est un facteur de pénalisation.

A l'aide du classifieur obtenu (disponible à l'adresse de la référence [10]), on peut tester ses propres pages, et avoir une vision très fine des facteurs pénalisants.

Un aperçu du classifieur

Il est important de noter que ces analyses dans le cadre du filtre Penguin peuvent être mises en place pour étudier n'importe quel autre phénomène, les outils de classification étant réellement génériques (ils sont utilisés en traitement des images, en marketing, en algorithmique des moteurs de recherche, pour lutter contre le spam dans les courriels, etc.).

En dehors de la confirmation d'un point déjà connu (les ancres optimisées sont un facteur de suroptimisation qu'il faut éviter), vous êtes maintenant sensibilisés à des outils nouveaux, plus évolués, et c'était bien là - et avant tout - notre but !

Références

[1] http://searchengineland.com/google-releases-first-penguin-update-year-206169
[2] http://seobserver.com/
[3] http://www.journaldunet.com/solutions/seo-referencement/penguin-3-0.shtml
[4] http://blog.seobserver.com/fr/64
[5] http://www.journaldunet.com/solutions/expert/58853/google-penguin-3-0---decryptage-concret-de-cette-mise-a-jour.shtml
[6] R. Quinlan: C4.5: *Programs for Machine Learning*, Morgan Kaufmann Publishers Inc., 1993.
[7]http://www.rulequest.com/see5-info.html
[8] Quinlan, J. R. (1986). *Induction of decision trees. Machine Learning*, 1(1):81-106.
[9] Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., ... & Steinberg, D. (2008). *Top 10 algorithms in data mining. Knowledge and Information Systems*, 14(1), 1-37. http://www2.cs.uh.edu/~ceick/ML/DM_Top10.pdf
[10]http://www.ix-labs.org/sylvain/result.txt

Guillaume Peyronnet *est gérant de Nalrem Médias.* **Sylvain Peyronnet** *est co-fondateur et responsable des ix-labs, un laboratoire de recherche privé. Ensemble, ils font des formations, pour en savoir plus : http://www.peyronnet.eu/blog/*