Utilisation de techniques de data mining pour le clustering d'URLs extraites de captures réseau de logiciels malveillants

Anthony VEREZ[1]

Soutenance de Projet de Fin d'Études, 2014

Plan

Introduction

Définitions

- Data Mining : Découvrir motifs cachés
- Machine Learning : Établir des prédictions

Introduction

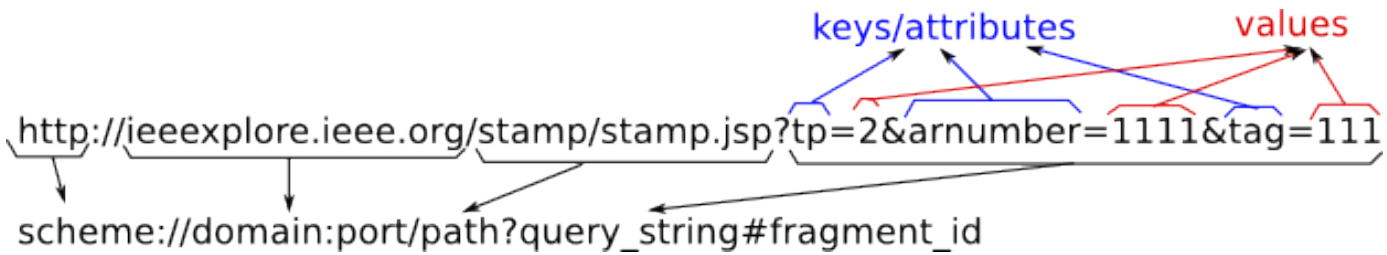
Définitions

- Data Mining : Découvrir motifs cachés
- Machine Learning : Établir des prédictions

Objectifs

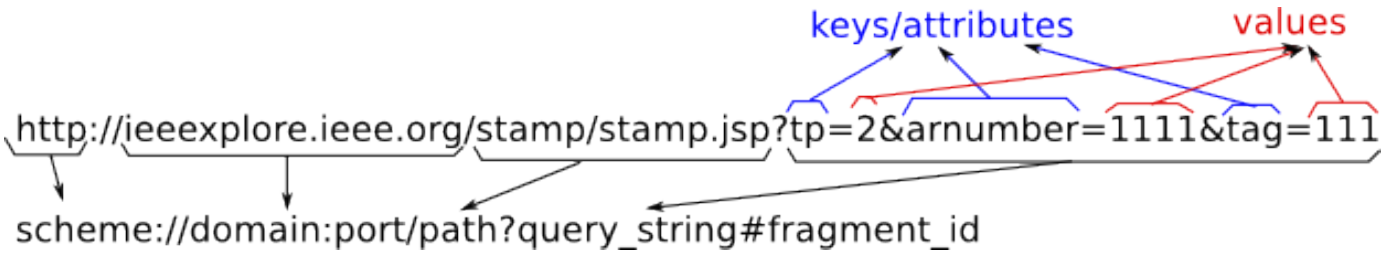
- Découvrir des familles de logiciels malveillants via URLs utilisées
- Créer des signatures d'URLs
- Automatisé, sans aucune information a priori

URL

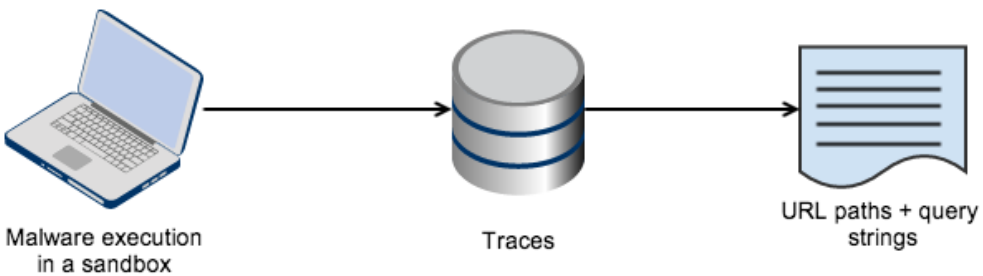


- Protocole de communication : HTTP
- Nom de domaine : on l'étudie pas
- Chemin
- Clés
- Valeurs
- Fragment : Aucun

URL



- Protocole de communication : HTTP
- Nom de domaine : on l'étudie pas
- Chemin
- Clés
- Valeurs
- Fragment : Aucun



Verbes HTTP

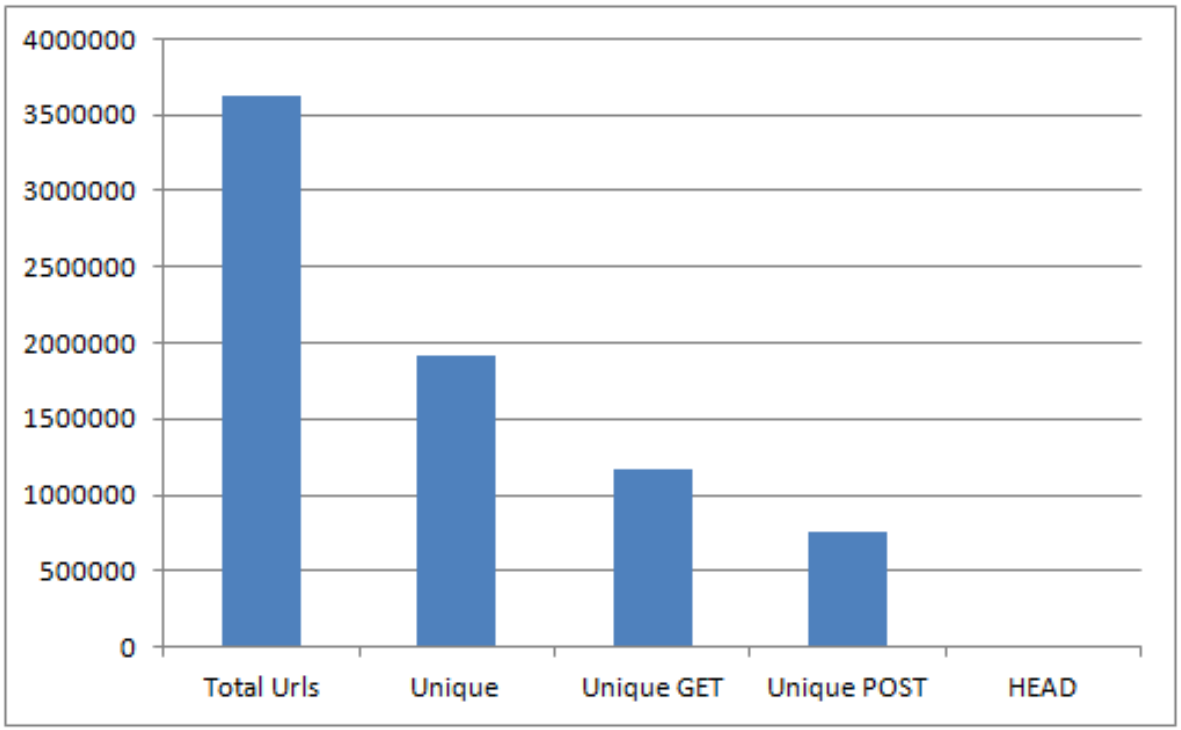
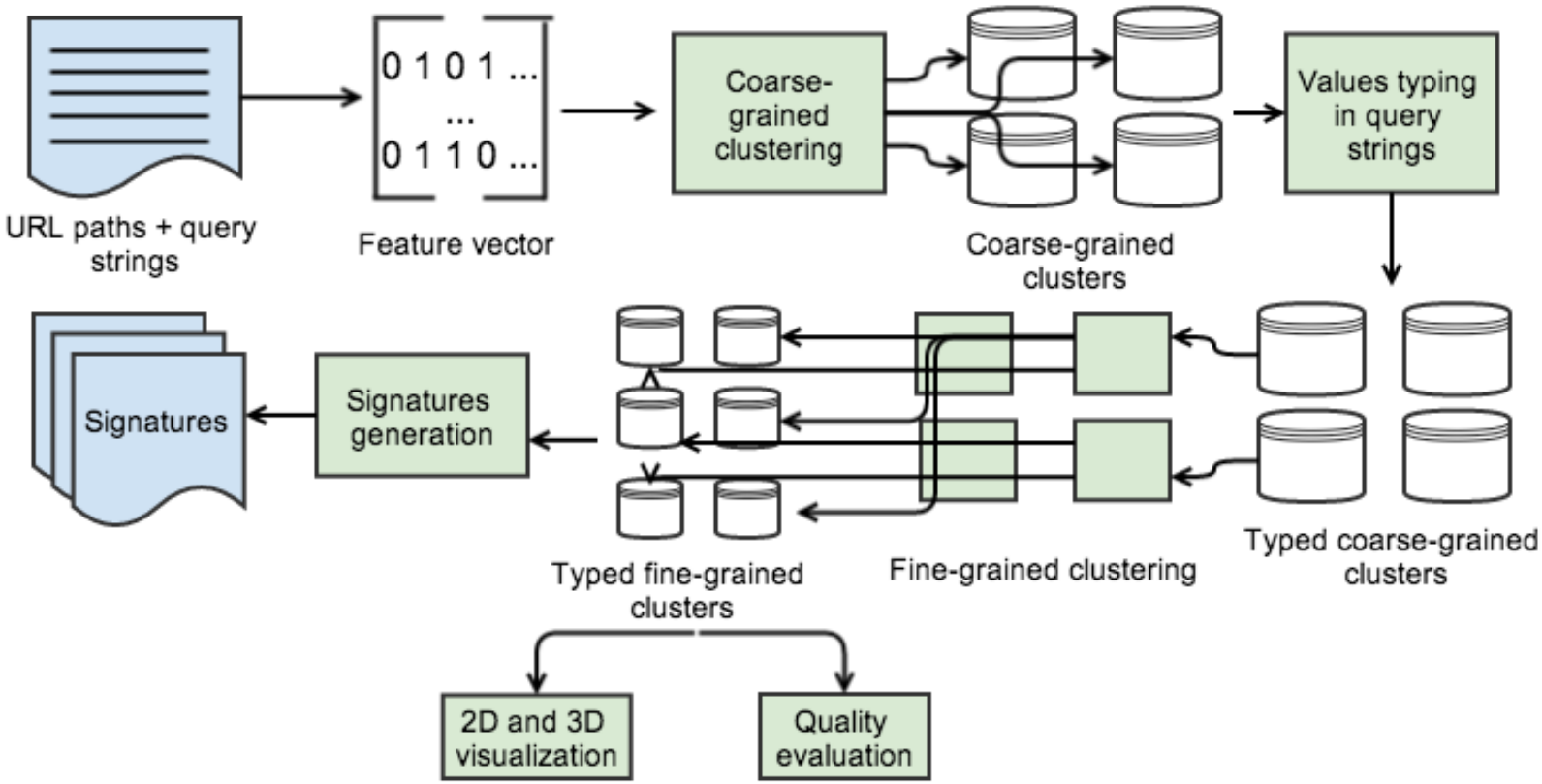


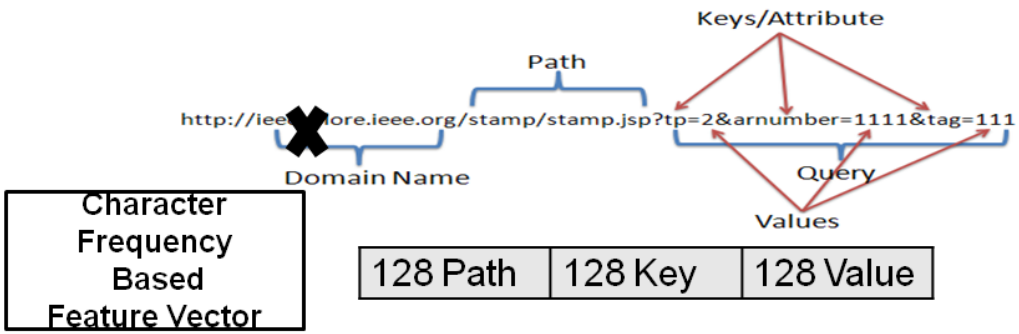
Figure: Requêtes HTTP par verbe pour notre dataset initial. On retient les requêtes GET.

Architecture et implémentation

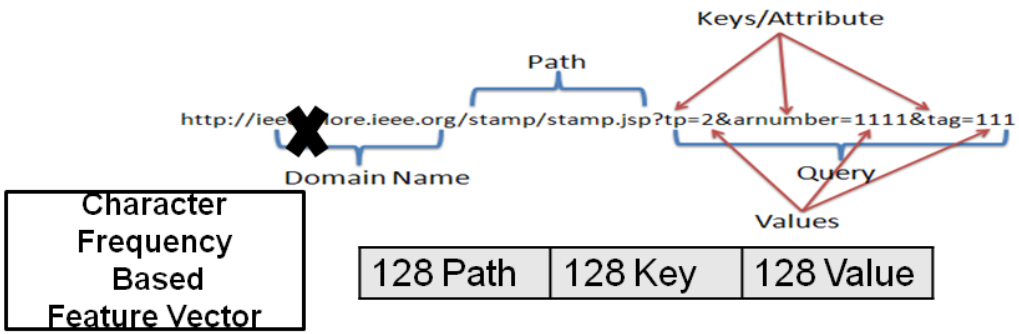
Architecture



Clustering grossier (coarse-grained)



Clustering grossier (coarse-grained)



Clustering avec K-means

- Objectif : diviser le dataset en plusieurs sous-datasets plus faciles à traiter
- Entrée : nombre de clusters
- Rapide
- Non supervisé

Typage des valeurs

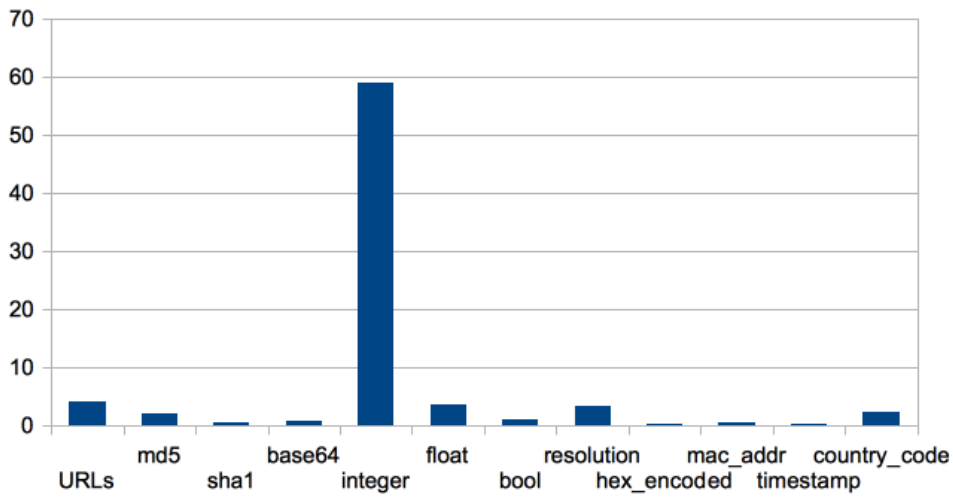
Objectifs :

- Préparer la génération de signatures
- Abstraire les valeurs variables

Typage des valeurs

Objectifs :

- Préparer la génération de signatures
- Abstraire les valeurs variables



MD5
SHA1
Base64
redirection URL
Réel
Entier
Booléen
Résolution
Nombre hexadécimal
adresse MAC
Chemin de fichier
Timestamp
Code pays

Clustering fin (fine-grained)

Objectif : Clustering fin donnant le résultat final

Clustering fin (fine-grained)

Objectif : Clustering fin donnant le résultat final
Distances

- Chemins : chaîne commune la plus grande
- v1 : Clés et valeurs séparées
- v2 : Couples clé/valeur

Clustering fin (fine-grained)

Objectif : Clustering fin donnant le résultat final

Distances

- Chemins : chaîne commune la plus grande
- v1 : Clés et valeurs séparées
- v2 : Couples clé/valeur

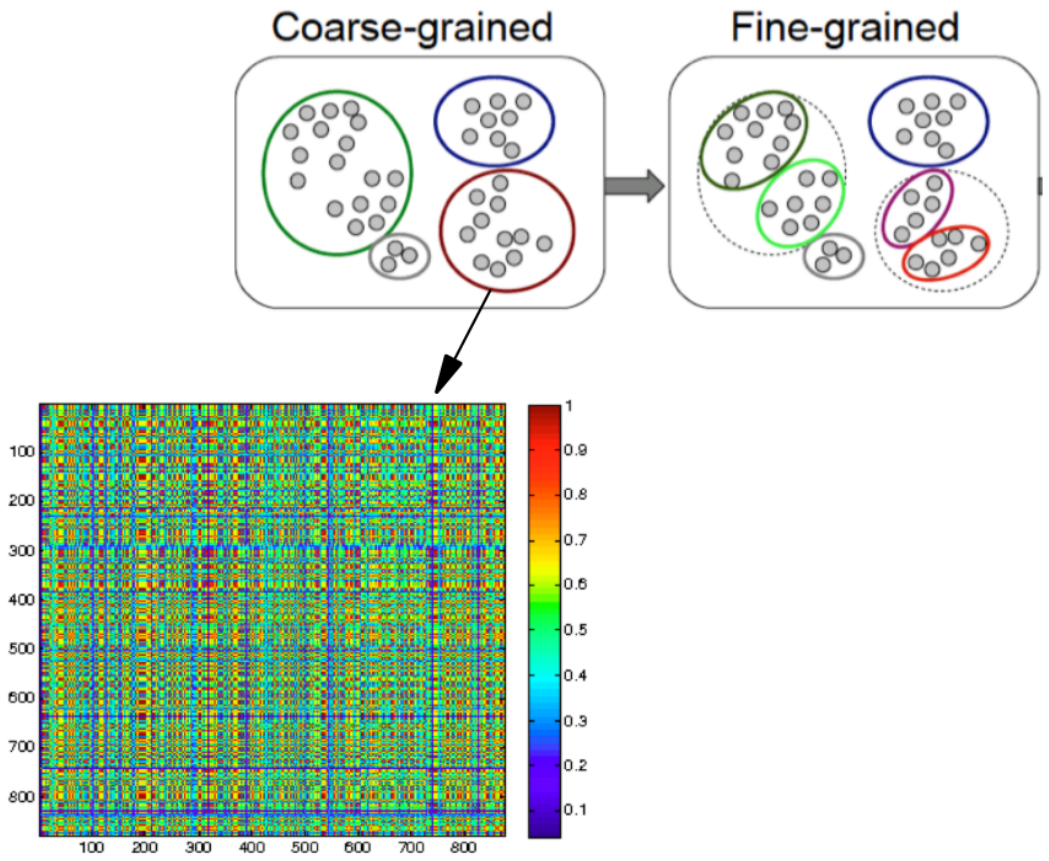
Clustering avec DBSCAN

- Densité
- Entrée : nombre points min. dans un cluster, distance minimum entre 2 objets d'un cluster
- Notion de bruit
- Faible complexité

Démo

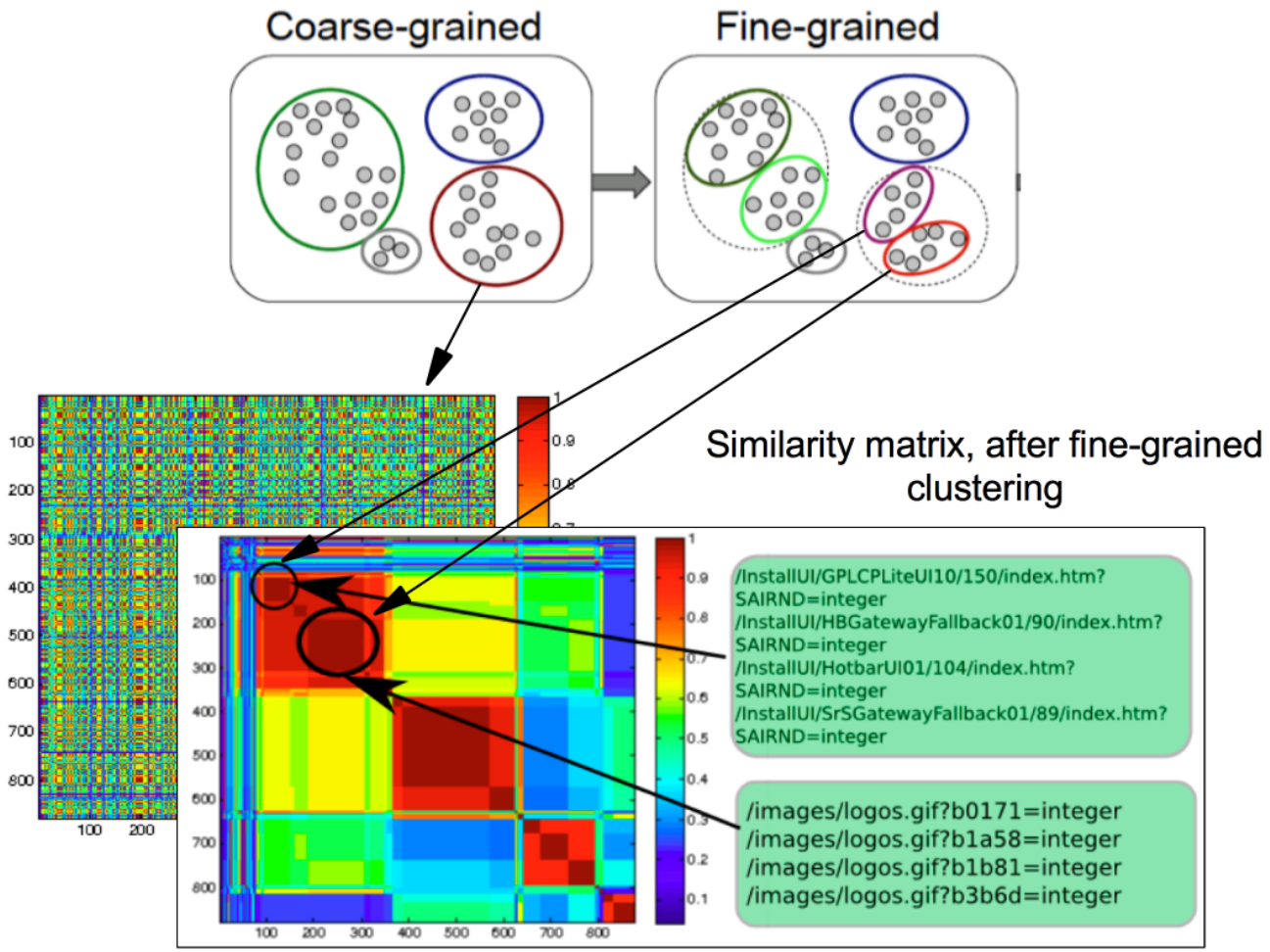
Résultats

Matrice de similarité



Similarity matrix, before fine-grained clustering

Matrice de similarité (2)



Quelques clusters

- /streamrotator/thumbs/ci/70449.jpg
- /streamrotator/thumbs/fm/167957.jpg
- /streamrotator/thumbs/hf/134852.jpg
- /streamrotator/thumbs/x/11857.jpg
- Signature : $.*\/streamrotator\/thumbs\/.*\/.*\/.*\.jpg.*$

- /InstallUI/GPLCPLiteUI10/150/index.htm?SAIRND=318453
- /InstallUI/HotbarUI01/104/index.htm?SAIRND=373046
- /generate/software/?SAIRND=490734
- /generate/software/?URLRND=354687
- Signature : $.*\/.*n.*e.*t.*\?(?=.*R.*ND\=[-\+]?[0-9]+).*$

Pistes d'améliorations

- Déploiement des signatures
- Datasets incrémentaux ou mode online
- Clustering grossier
- Typage

Conclusion

- Utiliser l'analyse d'anomalie pour créer des signatures
- Contributions
  - Datasets constitués uniquement d'URLs sans noms de domaine
  - Distances novatrices
  - Utilisation de clustering non supervisé avec DBSCAN
  - Création d'outils génériques pour analyse et visualisation pour des datasets de grande taille
- Remerciements

Distribution des fréquences des longueurs

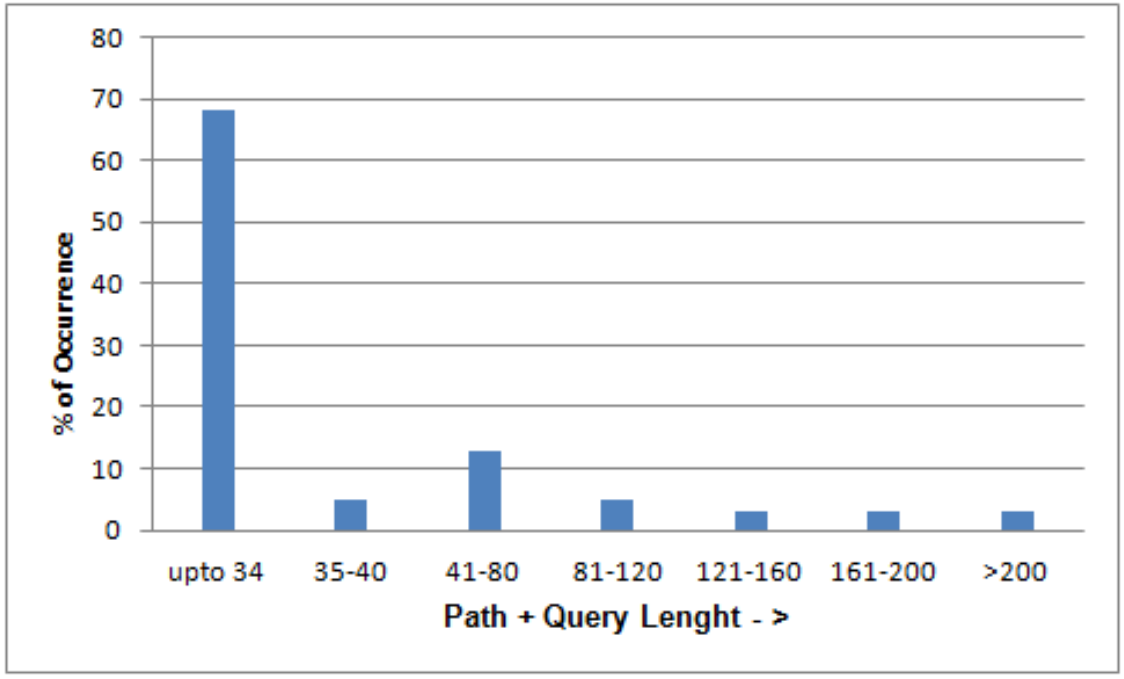
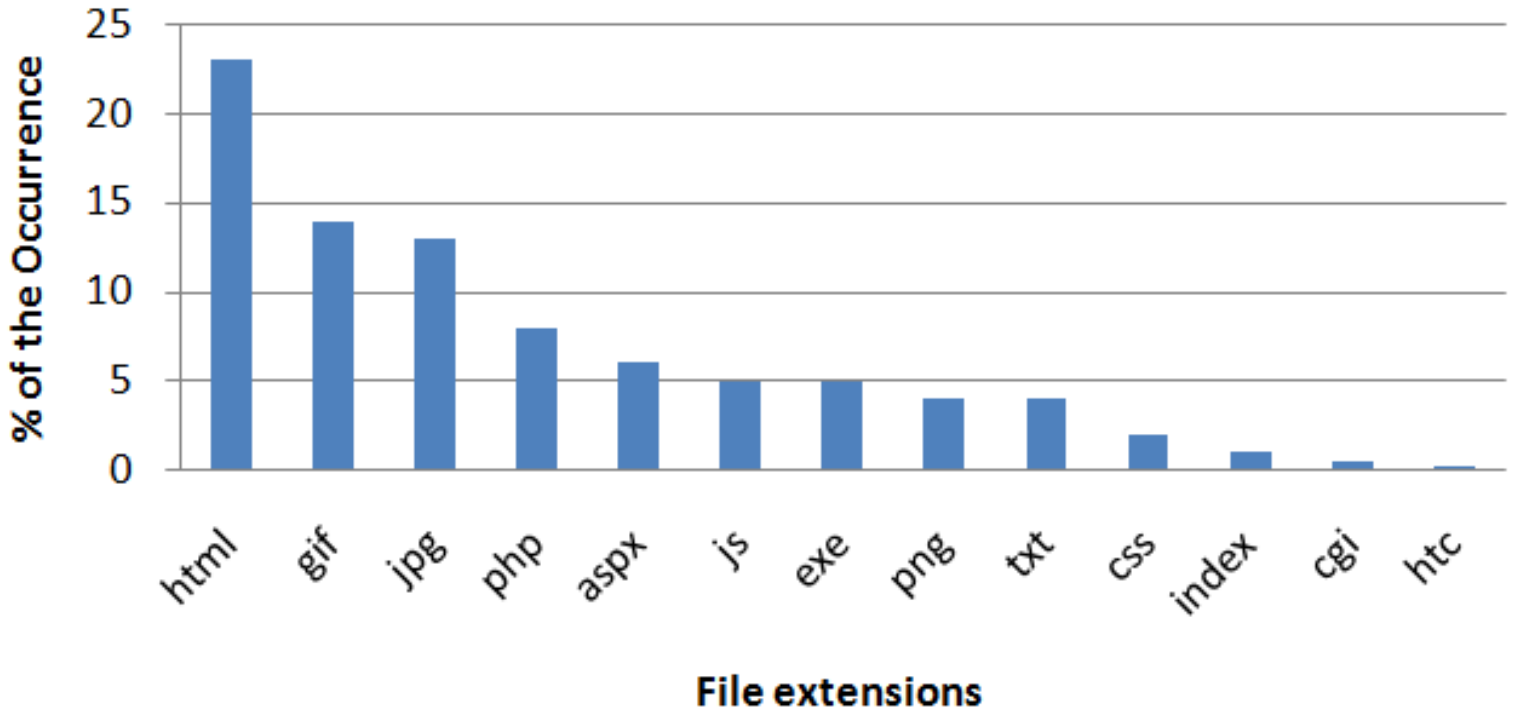


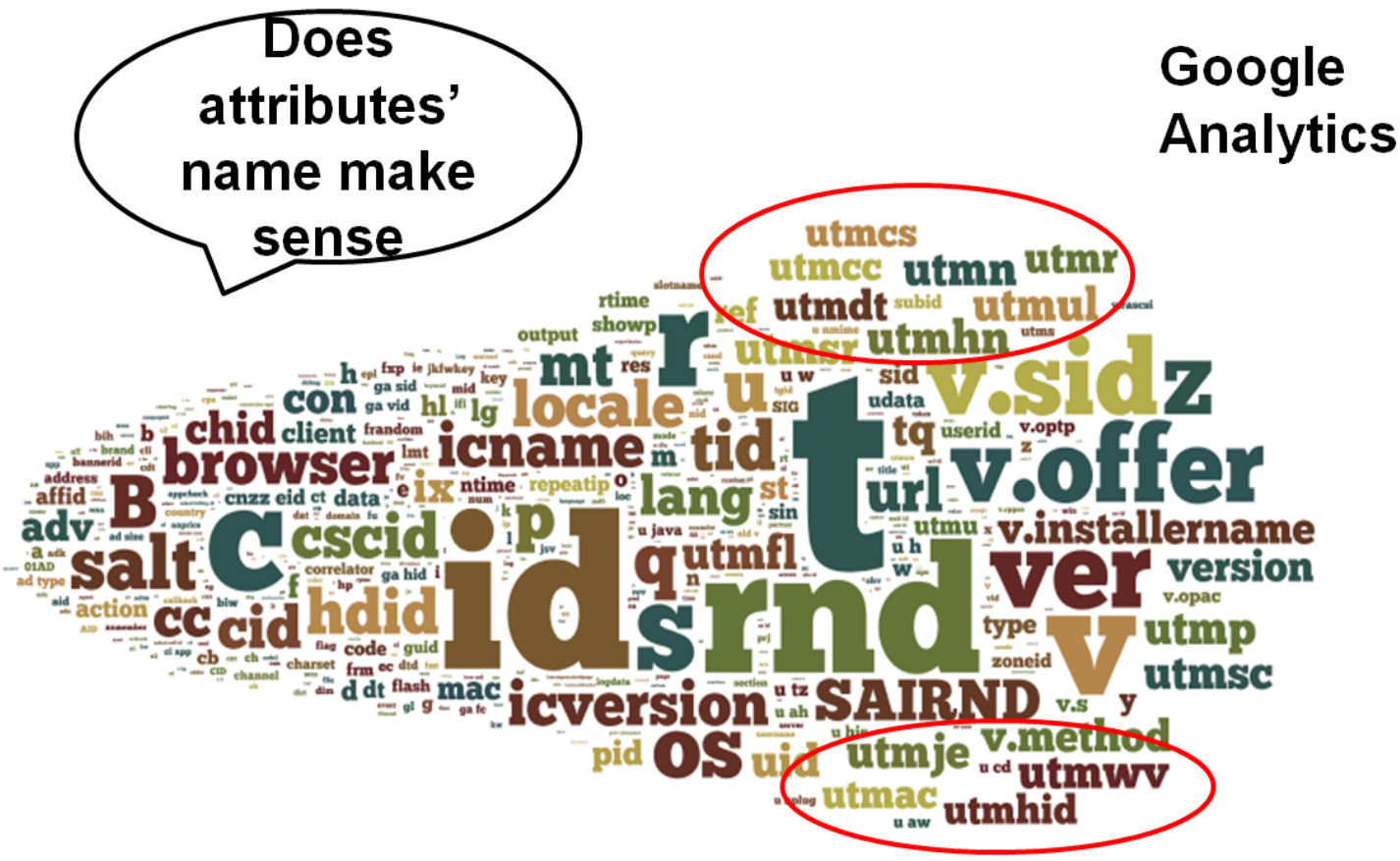
Figure: Distribution des fréquences des longueurs (chemin + clés + valeurs)

Extensions


% of the Occurrence
25
20
15
10
5
0
html
gif
jpg
php
aspx
js
exe
png
txt
css
index
cgi
htc
File extensions

Sac de mots

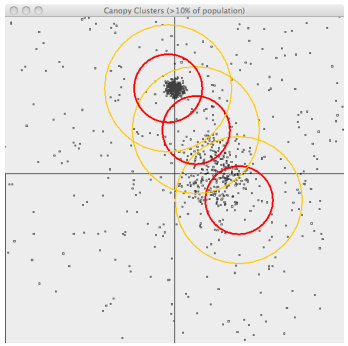
Cloud of Keys – How Query field can be used?



Quelques clusters (2)

- /utest/?jutr=68719&oo=2&7c820=3678e0&ra=0
- /utest/?jutr=9135&oo=2&9b4dd=43f20b&ra=0
- /utest/?jutr=44443&oo=2&b4253=4ed045&ra=0
- /utest/?jutr=74858&oo=2&7c85e=367a92&ra=0
- Signature :
  $.*\/utest\/\?(?=.*oo\=[-\+]?[0-9]+)(?=.*jutr\=[-\+]?[0-9]+)(?=.*ra\=[-\+]?[0-9]+).*$

Canopy Clustering



Start with a list of points and two distance thresholds $T1 > T2$.

1. Select a random point from this list to create a canopy center.
2. Compute its distance to all other points in the list.
3. Insert all the points which fall within the distance threshold of $T1$ into this canopy.
4. Remove from the main list all the points which fall within the threshold of $T2$. These points, already in a canopy, cannot be a canopy center or create new canopies.
5. Repeat from step 1 to 4 until the main list is empty.