UNIVERSITE DE MAROUA
FACULTE DE SCIENCES
DEPARTEMENT DE MATHÉMATIQUES - INFORMATIQUE

Web Mining

MASTER II

Avril 2024

Touza Isaac
isaac_touza@outlook.fr

Informations générales

- **Code UE** : DSI 553
- **Intitulé de L'UE** : Web Mining
- **Crédit** : -
- **Durée** : 30h
  - CM : 10h
  - TP : 10h
  - TPE : 5h
  - TD : 5h
- **Évaluations** :
  - CC (théorique ou pratique) : 2h ou 3h
  - Examen (théorique ou pratique) : 2h ou 3h
  - Rattrapage (théorique ou pratique) : 2h ou 3h

Références

- Bing Liu, Web Data Mining Exploring Hyperlinks, Contents, and Usage Data Second Edition, 2011
- Doru Tanasa, Brigitte Trousse, le prétraitement des fichiers logs Web dans le "Web Usage Mining« multi-sites. Équipe AxIS, INRIA Sophia Antipolis, 2004, Route des Lucioles, 06902 Sophia Antipolis Cedex, France
- Hanane Ezzikouri & Mohammed Erritali, Web Mining, Extraction des connaissances à partir des données du Web.
- HADJ-TAYEB Karima , Le Data Mining appliqué au WEB, 2018
- ZDR AVKO MARKOV AND DANIEL T. L AROSE , DATA MINING THE WEB, Uncovering Patterns in Web Content, Structure, and Usage , 2007

Objectifs du cours

Objectif général : Comprendre les principes fondamentaux, les techniques et les applications du web mining dans divers domaines.

Objectifs spécifiques :

- Comprendre les concepts fondamentaux du web mining.
- Maîtriser les méthodes de collecte de données sur le web.
- Appliquer les techniques d'extraction d'information à partir de données web.
- Analyser les données web à l'aide de différentes méthodes.
- Utiliser des techniques d'apprentissage automatique pour le web mining.
- Illustrer les applications pratiques du web mining dans différents domaines.
- Sensibiliser aux questions éthiques et de confidentialité liées au web mining.
- Réaliser des travaux pratiques pour mettre en pratique les connaissances acquises.

Près-requis et consignes

Près-requis

- Familiarité avec les concepts de base en statistiques et en mathématiques.
- Connaissances du langage Python
- Théorie de graphe

Consignes

- Assister à tous les cours
- Être attentifs et actifs pendant le cours
- Faire tous les exercices de TD et TP
- Refaire plusieurs fois les exercices d'applications
- **Ne manquez jamais un TP**

Plan du cours I

1. **Chapitre 1 : Introduction au Web Mining**
   - Introduction
   - Data Mining : définitions et tâches
   - Web Mining : objectifs
   - Axes du Web Mining
   - Processus du Web Mining
   - Types de données du Web
   - Techniques du Web Mining
   - Domaines d'applications
2. **Chapitre 2 : Les Méthodes du Web Mining**
   - ID3
   - C4.5
   - Knn(k-nearest neighbor)
   - Bayes naïf
   - Apriori

Plan du cours II

- K-means

3 Chapitre 3 : Le Web Content Mining

- Introduction
- Définition
- Collecte des données du Web
- Pré-traitement des données textuelles
- Pondérations ou calcul des fréquences
- Étiquetage du document
- Extractions des termes
- Application : Classification de texte

4 Chapitre 4 : Le Web Structure Mining

- Introduction
- Terminologie du Web structure Mining
- Structures du Web

Plan du cours III

- Techniques d'analyses d'hyperliens
- Algorithmes de Web Structure Mining

5 **Chapitre 5 : Le Web Usage Mining**

- Introduction
- Définitions des concepts
- Les fichiers Logs
- Processus du Web Usage Mining

CHAPITRE 1

Introduction au Web Mining

Introduction

- L'accroissement exponentiel des utilisateurs d'Internet, entraine la massification des données, et l'accès à des informations utiles et pertinentes se trouvant sur internet devient donc de plus en plus complexe.
- Par ailleurs cette augmentation massive des données sur internet crée chez les utilisateurs plusieurs besoins (classification , recherche d'information, relation entre objets...)
- Ainsi, des outils performants devront être mis à la disposition des utilisateurs du Web et des entreprises pour répondre de manière performante et intelligente à leurs besoins : d'où la naissance du **Web Mining**

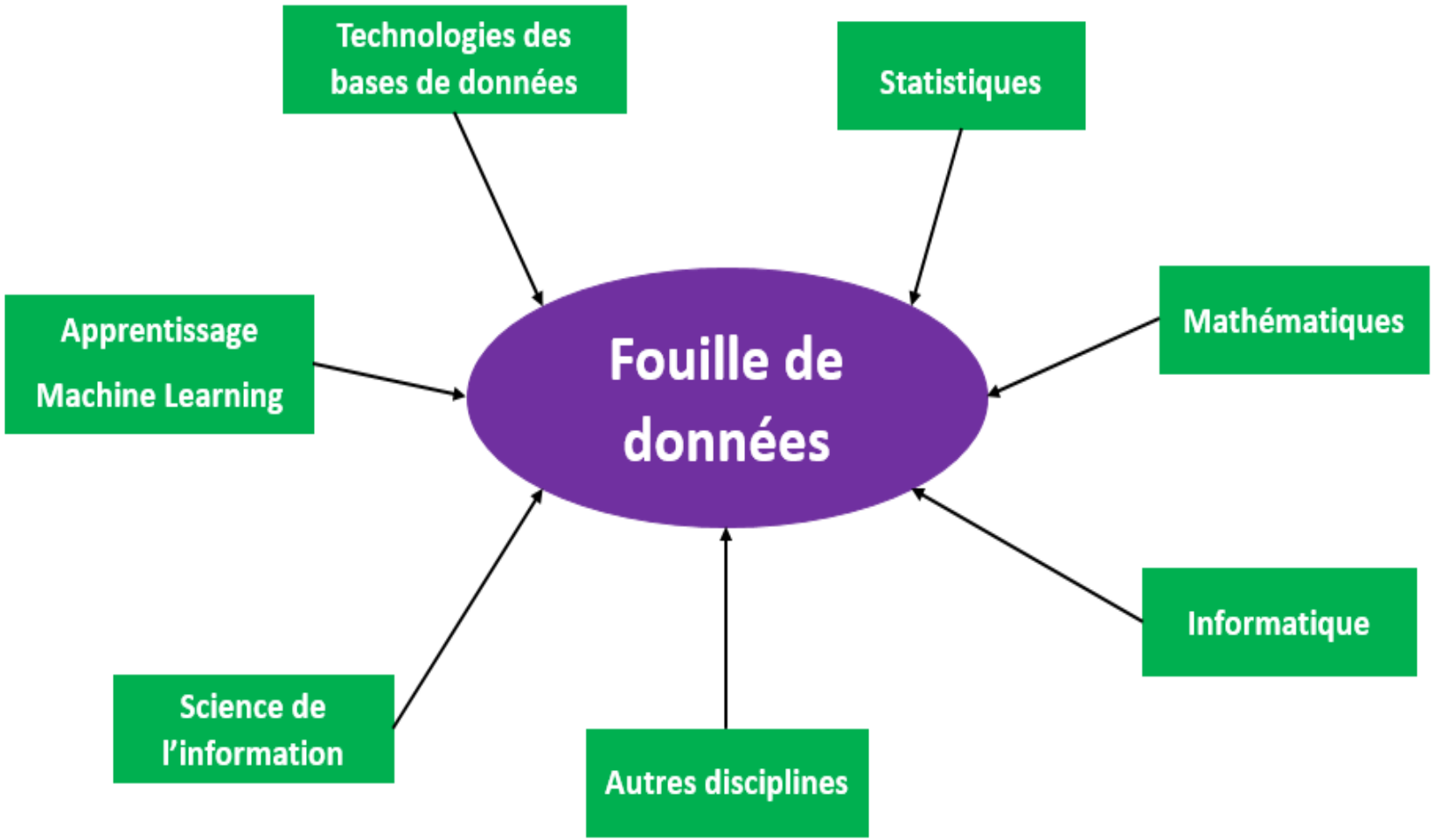
Définitions

data mining ou fouille de données

- Extraction d'informations intéressantes (non triviales, implicites, probablement inconnues,et potentiellement utiles) à partir d'une grande bases des données
- Ensemble de techniques d'exploration de données permettant d'extraire d'une base de données des connaissances sous la forme de modèles de description afin de décrire le comportement actuel des données et/ou prédire le comportement futur des données.

Le **data mining** se sert des techniques et méthodes , mathématiques, statistiques et algorithmiques pour extraire une connaissance ou une décision à partir des données élémentaires disponibles dans une entrepôt des données (data warehouse ou big data)ou d'une base des données quelconque : On parle de **(Knowledge Discovery in Databases – KDD)**

Définitions



Tâches du data mining

1. **Classification** : on examine les caractéristiques d'un nouvel objet pour l'affecter à une classe prédéfinie.
2. **Estimation** : Elle porte sur des variables continues(numérique) et établit le lien entre une combinaison de critères
3. **Segmentation** : il s'agit de déterminer quelles observations vont naturellement ensemble sans privilégier aucune variable.
4. **Prédiction** : cette fonction est proche de la classification ou de l'estimation, mais les observations sont classées selon un comportement ou une valeur estimée futurs.
5. **Recherche d'association** : Elle permet de rechercher et à découvrir dans une grande base des données les relations et/ou les règles cachées qu'il existe entre les attributs (variables) et qu'ils sont utiles pour la prise de décision.

Pourquoi le Web Mining ?

Quelques raisons d'être du Web Mining sont :

- **L'explosion des données** : Les outils de collecte automatique des données et les bases de données conduisent à d'énormes masses de données stockées dans des entrepôts
- **Submergés par les données, manque de connaissance !** :
- **Données en trop grandes quantités pour être traitées manuellement ou par des algorithmes classiques** : Nombre d'enregistrements en million ou milliard, Donnée de grande dimension (trop de champs/attributs/caractéristiques), Sources de données hétérogènes
- **Nécessité économique** : e-commerce, Haut degré de concurrence, personalisation, fidélisation de la clientèle, market segmentation

Web Mining : objectifs

Lorsque le **data mining** est utilisée pour d'exploration de données dans les ressources d'internet ou les données le concernant, alors on parlera de la **fouille des données du web** ou **le web mining**

Le Web Mining :

- Ensemble des techniques de data mining appliquées aux données du web.
- Application des techniques du data mining pour l'extraction d'informations pertinentes à partir des ressources disponibles dans le Web.

Le web mining présente deux principaux objectifs que sont :

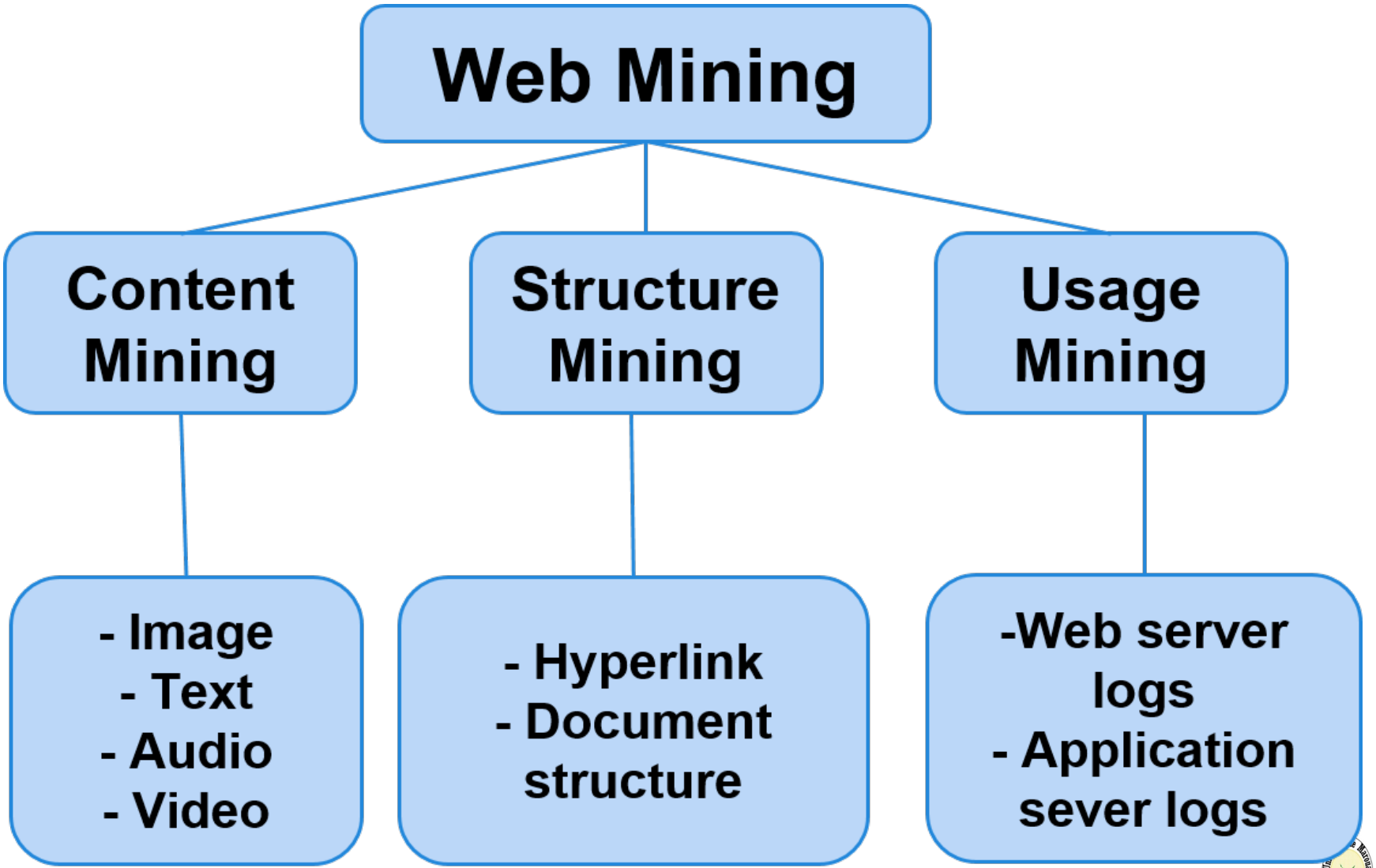
1. **L'amélioration et la valorisation des sites Web**
2. **La personnalisation**

Axes du Web Mining I

Selon ces cibles, la fouille du web peut être divisée en trois types :

1. la fouille de l'usage du web : **Web Usage mining**
2. la fouille du contenu du web : **Web Content mining**
3. la fouille de la structure du web : **Web Struture mining**

Axes du Web Mining II



Processus du Web Mining

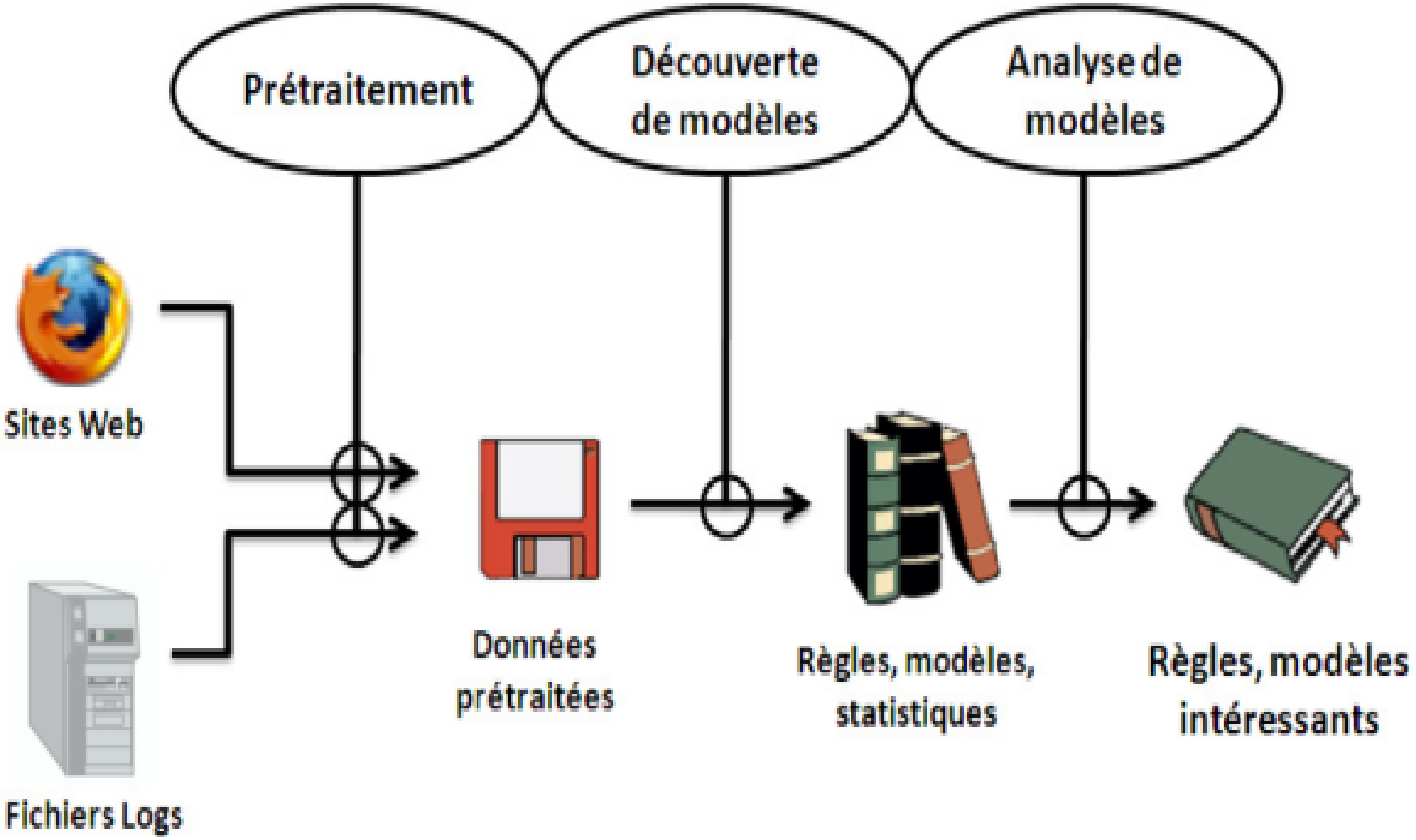
Le processus du Web Mining se déroule en quatre étapes telles que décrites ci-dessous :

1. **Collecte des données** : La première étape est la collecte des données à exploiter qui consiste à rassembler les données qui vont être analysées.
2. **Le Pré-traitement des données collectées** : Elle consiste à nettoyer les données et à les transformer. Le pré-traitement a comme objectif la structuration et l'amélioration de la qualité des données contenues dans les fichiers pour les préparer à une analyse des usagers.
   - La phase de **nettoyage** des données consiste à filtrer les données inutiles à travers la suppression des requêtes auxiliaires et invalides ne faisant pas l'objet de l'analyse effectuée et celle provenant des robots web.
   - La phase de **transformation** des données consiste à formater les données afin de les rendre facilement exploitable.

Processus du Web Mining

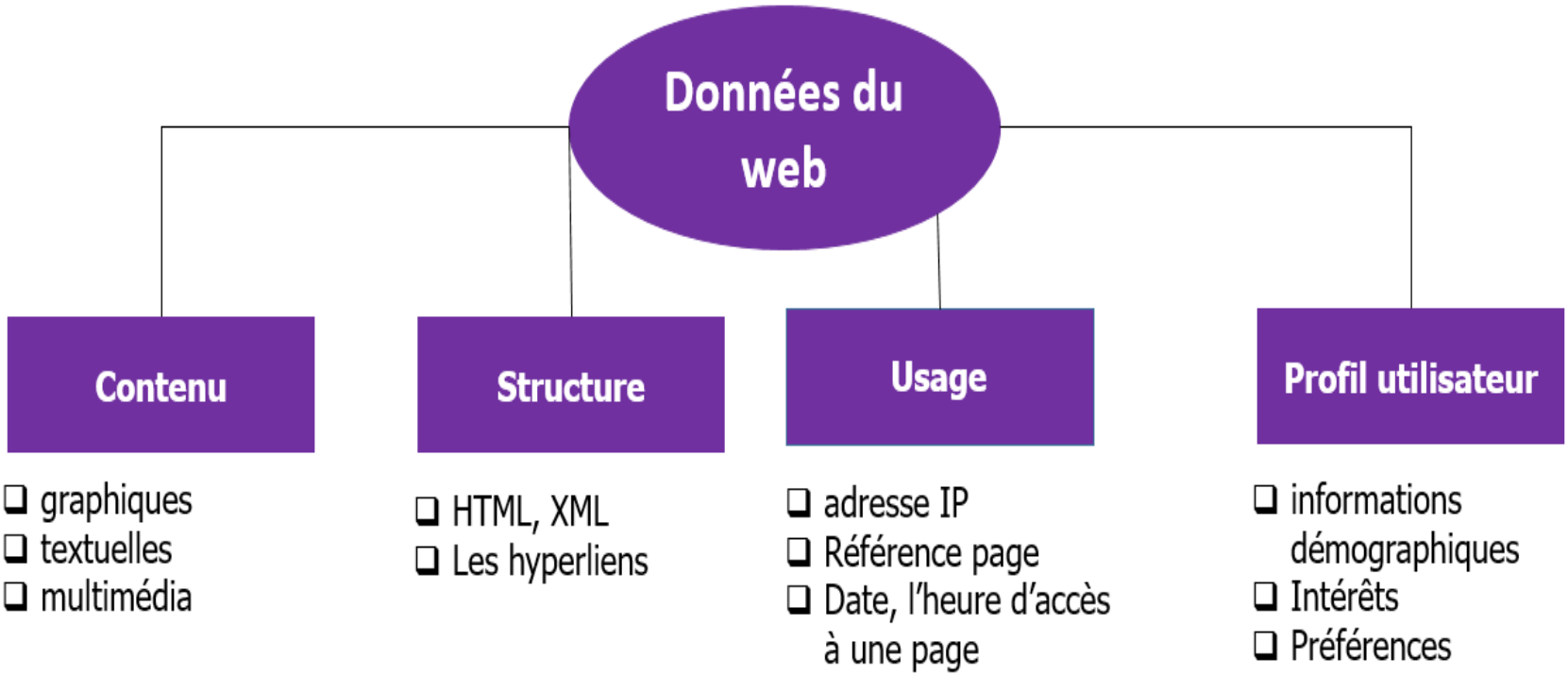
3. **Extraction d'informations** : Elle permet d'appliquer une des techniques d'extraction pour la découverte des motifs. Cette découverte s'appuie sur des méthodes et des algorithmes développés à partir de plusieurs domaines tels que les statistiques, le Data Mining, l'apprentissage machine et la reconnaissance des formes.
4. **Analyse et présentation à l'utilisateur d'un contenu ciblé.** : L'analyse et l'interprétation des motifs découverts s'effectue à travers les techniques de visualisation. Cette analyse nécessite le recours à un ensemble d'outils pour ne garder que les résultats les plus pertinents et les plus significatifs. Elle est considérée comme une étape importante du processus d'extraction, car une fois les motifs trouvés, il faut être en mesure de bien sélectionner les motifs intéressants et de pouvoir les valider.

Processus du Web Mining



Les types de données du Web

Les types de données exploitées en data Mining sont :



Techniques du Web Mining

Ils permettent d'accomplir des analyses qui peuvent être regroupées en deux catégories :

1. **Les techniques descriptives :**consiste à trouver les caractéristiques générales relatives aux données fouillées .
   - **Classification**
     - ID3
     - C4.5
     - Classifieur Bayésien
     - Knn(k-nearest neighbor)
   - **Association**
     - Apriori
     - FP-Growth
     - Eclat
     - OPUS

Techniques de Web Mining

2. **Les techniques prédictives :** Consiste à utiliser certaines variables pour prédire les valeurs futures inconnues de la même variable ou d'autres variables.
   - **Estimation**
     - EM
     - EDA(Estimation of Distribution Algorithms)
     - SEM(Stochastique,Estimation,Maximisation)
   - **Clustering**
     - k-means
     - EM (Maximisation d'espérance)
     - OPTICS
     - BDSCAN
   - **Prévision**
     - Regression logistique
     - knn(K plus proches voisins)
     - Bayes naïf
     - SVM(Machine à support de vecteur)

Domaines d'applications

- **Domaine des assurances :**
  - analyse des risques (caractérisation des clients à hauts risques, etc.)
  - automatisation du traitement des demandes (diagnostic des dégâts et détermination automatique du montant des indemnités)
- **Services financiers :**
  - Attribution de prêts automatisés, support à la décision de crédit
  - Détection de fraude
  - Marketing ciblé
- **Grande distribution :**
  - profils de consommateurs et modèles d'achats
  - marketing ciblé

Domaines d'applications

- **Médecine :**
  - Aide au diagnostic
- **Ticket de caisse :**
  - Liste des achats.
  - Heure de passage en caisse.
  - Quels sont les articles le plus souvent achetés ensemble ?
  - Promotions groupées, agencement du magasin . .

CHAPITRE 2

Les techniques du Web Mining

L'algorithme ID3

Présentation

L'Algorithme ID3 a été développé à l'origine par Ross Quinlan. Il a tout d'abord été publié dans le livre "Machine Learning" en 1986. C'est un algorithme de classification supervisé, c'est-à-dire qu'il se base sur des exemples déjà classés dans un ensemble de classes pour déterminer un modèle de classification. Le modèle que produit ID3 est un arbre de décision. Cet arbre servira à classer de nouveaux échantillons.

L'algorithme ID3

Principe

1. Chaque exemple en entrée pour cet algorithme est constitué d'une liste d'attributs.
2. Un de ces attributs est l'attribut « **cible** » et les autres sont les attributs « non cibles ».On appelle aussi cette "**cible**" la "classe".
3. L'arbre de décision va permettre de prédire la valeur de l'attribut « cible » à partir des autres valeurs.
4. ID3 construit l'arbre de décision récursivement. À chaque étape de la récursion, il calcule parmi les attributs restant pour la branche en cours, celui qui maximisera le gain d'information. C'est-à-dire l'attribut qui permettra le plus facilement de classer les exemples à ce niveau de cette branche de l'arbre.
5. On appelle ce calcul **l'entropie de Shannon**

L'algorithme ID3

Le calcul de **l'entropie de Shannon** est fait en utilisant la formule :

$$\textbf{Entropie(S)} = -\sum_{c \in \text{classes(S)}} P_c \times log_2(P_c)$$

- S est le set d'exemples.
- La quantité minimale de données redondantes à ajouter pour qu'un message ayant une probabilité p d'arriver sans être corrompu est $-log_2(P)$
- P_c est la proportion d'exemples de S ayant pour classe résultante c.

ID3

En bref, voila comment fonctionne l'algorithme ID3 :

1. Calculer l'entropie de tous les attributs en utilisant l'ensemble d'apprentissage S
2. Partitionner l'ensemble S en utilisant l'attribut pour lequel l'entropie est minimum (gain d'information maximum)
3. Construire le noeud de l'arbre avec cet attribut
4. Recommencer récursivement sur chaque sous arbre avec chaque sous-ensemble.

Le pseudo-code de l'algorithme ID3 est donné ci-dessous :

ID3 : Pseudo-code

Algorithme 1 ID3

Entrée(s): exemples(liste d'exemples étiquetés), questions(liste des attributs non utilisés jusqu'à présent) ;

Sorties(s): un nœud

```
Si exemples est vide alors
    Finir la fonction sans construire de nœud
Fin Si
Si tous les exemples sont la même classe alors
    Retourner une feuille ayant cette classe
Fin Si
Si questions est vide alors
    Retourner une feuille avec la classe la plus fréquente
Fin Si
q = attribut optimal (avec le plus grand gain d'entropie)
n = nouveau nœud créé qui testera l'attribut q
Pour chaque v = valeur possible de q faire
    e = l'ensemble des éléments de exemples ayant v comme valeur à l'attribut q
    chaque nœud fils de n est créé par ID3(e, questions\{q})
Fin pour
Retourner n
```

ID3 : Exemple

Considérons les données du tableau ci-dessous :

Jour	**Attributs des exemples**				**Classe**
Jour	Prévisions	Température	Humidité	Vent	Jouer
1	Ensoleillé	Chaud	Élevée	Faible	Non
2	Ensoleillé	Chaud	Élevée	Fort	Non
3	Nuageux	Chaud	Élevée	Faible	Oui
4	Pluvieux	Moyen	Élevée	Faible	Oui
5	Pluvieux	Frais	Normale	Faible	Oui
6	Pluvieux	Frais	Normale	Fort	Non
7	Nuageux	Frais	Normale	Fort	Oui
8	Ensoleillé	Moyen	Élevée	Faible	Non
9	Ensoleillé	Frais	Normale	Faible	Oui
10	Pluvieux	Moyen	Normale	Faible	Oui
11	Ensoleillé	Moyen	Normale	Fort	Oui
12	Nuageux	Moyen	Élevée	Fort	Oui
13	Nuageux	Chaud	Normale	Faible	Oui
14	Pluvieux	Moyen	Élevée	Fort	Non

ID3 : Exemple

Prévisions, **Température**, **Humidité** et **Vent** sont les quatre attributs qui déterminent chacun des exemples qui vont être fournis. On peut voir qu'il existe uniquement 36 exemples différents pour cette configuration d'attributs :

$$\{\text{Ensoleillé,Nuageux,Pluvieux}\} \times \{\text{Chaud,Moyen,Frais}\} \times \{\text{Élevée,Normale}\} \times \{\text{Faible,Fort}\} = 3 \times 3 \times 2 \times 2 = 9 \times 4 = 36$$

Dans cet exemple, les seules classes possibles sont **Oui** et **Non**. Autrement dit la classification de la cible est "**devrions-nous jouer au tennis ?**" qui peut être **oui** ou **non**.
Donc l'entropie vaut :

$$-P_{Oui} \times log_2(P_{Oui}) - P_{Non} \times log_2(P_{Non})$$

ID3 : Exemple

Initialement, l'algorithme prend tout le set S=$\{J_1, J_2, J_3, ..., J_{14}\}$. Et comme 9 des 14 exemples donnent la réponse (ou classe) **Oui** et 5 sur 14 donnent la réponse (ou classe) **Non**,

$$P_{Oui} = \frac{9}{14}$$
$$P_{Non} = \frac{5}{14}$$

On peut donc calculer que :

$$Entropie(S) = -(\frac{9}{14}) \times log_2(\frac{9}{14}) - (\frac{5}{14}) \times log_2(\frac{9}{14}) = 0.94$$

ID3 : Exemple

Calculons le gain d'entropie pour chacun des attributs :
Rappelons que ce gain est donnée par la formule :

$$\text{Gain(S,A)} = \text{Entropie(S)} - \sum_{v \in valeurs(A)} \frac{|S_v|}{|S|} \times \text{Entropie}(S_v)$$

ID3 : Exemple

Prévisions

L'attribut **Prévisions** a trois valeurs possibles : Ensoleillé,Nuageux et Pluvieux

Gain(S,Prévisions) = Entropie(S) - $\frac{5}{14}\times$ Entropie($S_{Ensoleillé}$) - $\frac{4}{14}\times$ Entropie($S_{Nuageux}$) - $\frac{5}{14}\times$ Entropie($S_{Pluvieux}$)

$$
\begin{aligned}
&= 0{,}94 - \tfrac{5}{14} \times \left(-\tfrac{3}{5} log_2\left(\tfrac{3}{5}\right) - \tfrac{2}{5} log_2\left(\tfrac{2}{5}\right)\right) - \tfrac{4}{14} \times \\
&\left(-\tfrac{0}{5} log_2\left(\tfrac{0}{4}\right) - \tfrac{4}{4} log_2\left(\tfrac{4}{4}\right)\right) - \tfrac{5}{14} \times \left(-\tfrac{3}{5} log_2\left(\tfrac{3}{5}\right) - \tfrac{2}{5} log_2\left(\tfrac{2}{5}\right)\right) \\
&= 0{,}94 - 0{,}357 \times (0{,}97) \\
&= 0{,}24742
\end{aligned}
$$

Donc **Gain(S,Prévisions) = 0,24742**

ID3 : Exemple

En procédant de la même manière avec les calculs de gain d'attributs des autres variables, nous obtenons :

Gain(S,Vent) = Entropie(S) - $\frac{8}{14}\times$ **Entropie(**S_{Faible}**) -** $\frac{6}{14}\times$ **Entropie(**S_{Fort}**) = 0.048**

Gain(S,Humidité) = Entropie(S) - $\frac{8}{14}\times$ **Entropie(**$S_{\text{É}levée}$**) -** $\frac{6}{14}\times$ **Entropie(**$S_{Normale}$**)=0.153**

Gain(S,Température) = Entropie(S) - $\frac{4}{14}\times$ **Entropie(**S_{Chaud}**) -** $\frac{6}{14}\times$ **Entropie(**S_{Moyen}**) -** $\frac{4}{14}\times$ **Entropie(**S_{Frais}**) = 0.028**

ID3 : Exemple

Récapitulons :

Gain(S,Température) $<$ Gain(S,Vent) $<$ Gain(S,Humidité)$<$ Gain(S,Prévisions).
On peut voir que le plus grand gain est pour **Prévisions**. C'est donc Prévisions qui est le premier attribut testé dans l'arbre c'est à dire il est l'attribut de décision dans le nœud racine de notre arbre.

Nous obtenons donc l'arbre temporaire ci-dessous :

ID3 : Exemple

Ensoleillé

Jour	Température	Humidité	Vent	jouer
1	Chaud	Elevée	Faible	Non
2	Chaud	Elevée	Fort	Non
8	Moyen	Elevée	Faible	Non
9	Frais	Normale	Faible	Oui
11	Moyen	Normale	Fort	Oui

?

Nuageux

Jour	Température	Humidité	Vent	jouer
3	Chaud	Elevée	Faible	Oui
7	Frais	Normale	Fort	Oui
12	Moyen	Elevée	Fort	Oui
13	Chaud	Normale	Faible	Oui

Jouer = Oui

Pluvieux

Jour	Température	Humidité	Vent	jouer
4	Moyen	Elevée	Faible	Oui
5	Frais	Normale	Faible	Oui
6	Frais	Normale	Fort	Non
10	Moyen	Normale	Faible	Oui
14	Moyen	Elevée	Fort	Non

?

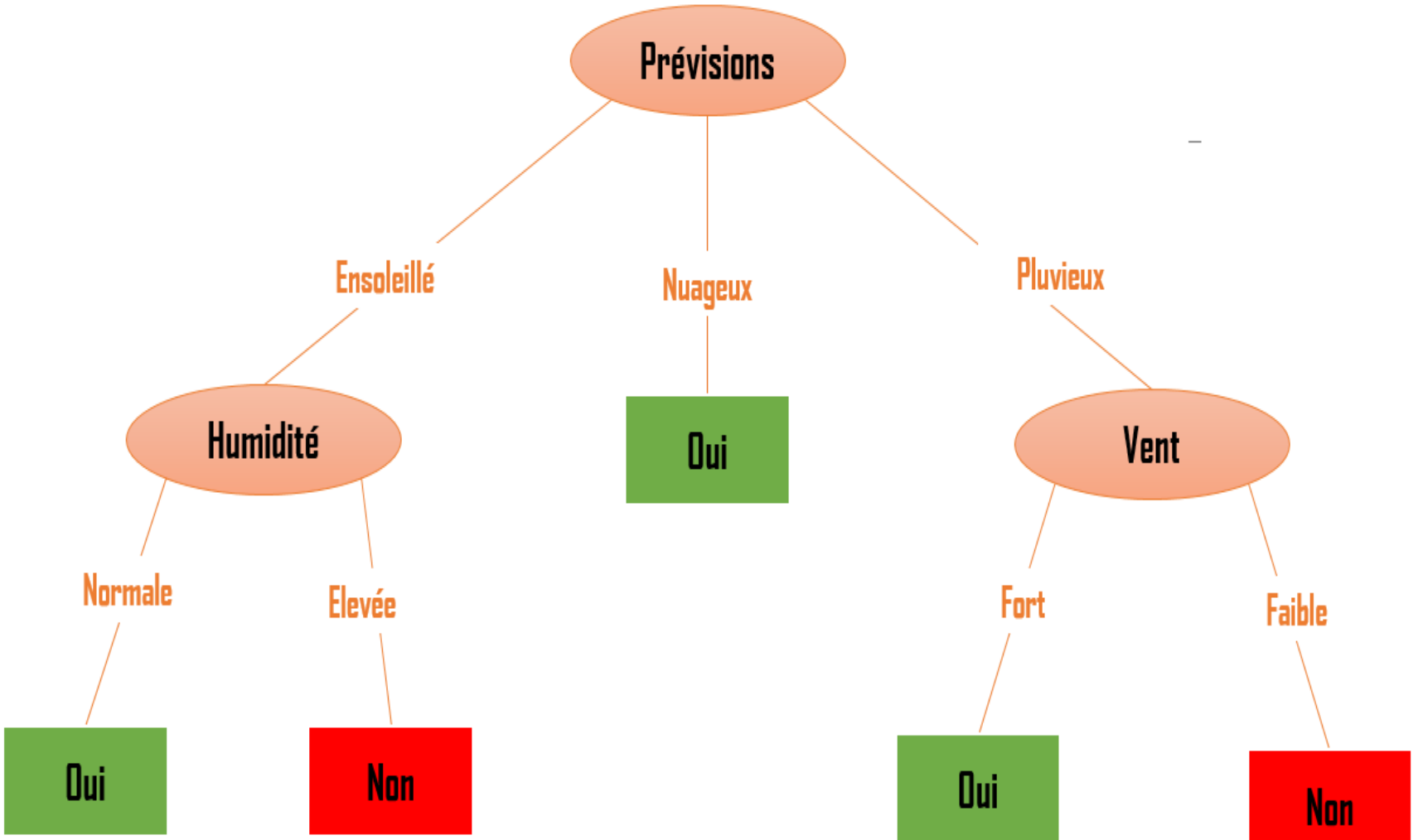
ID3 : Exemple

Il faut maintenant continuer à mettre des nœuds après **Ensoleillé** et **Pluvieux** car tous les exemples ne donnent pas le même résultat.

Il faut donc à cet effet déterminer pour **Ensoleillé** et **Pluvieux** le meilleur attribut à tester en utilisant à nouveau le gain. Cependant il n'est plus utile de tester le gain de Prévisions étant donné qu'il vient d'être utilisé.

. L'arbre de décision finale obtenu par ID3 donné ci-dessous :

ID3 : Exemple


Prévisions
Ensoleillé
Nuageux
Pluvieux
Humidité
Oui
Vent
Normale
Elevée
Fort
Faible
Oui
Non
Oui
Non

C4.5

Présentation

L'algorithme C4.5 utilisé pour générer un arbre de décision développé a été proposé en 1993, toujours par Ross Quinlan, pour pallier les limites de l'algorithme ID3 vu précédemment.Il est classé numéro 1 dans le top 10 des algorithmes dans le domaine de l'exploration de données publié par Springer LNCS en 2008.

C4.5 : Améliorations

Quelques extensions à ID3 introduit par l'algorithme C4.5 sont :

1. **Gestion des attributs de valeur inconnue**
2. **Gestion des attributs à valeur sur intervalle (attributs continus)**
3. **L'élagage de l'arbre de décision après création**
4. **Gestion des données d'apprentissage avec des valeurs d'attribut manquantes**
5. **Gestion des attributs avec des coûts différents**.

En bref, C4.5 est une amélioration d'ID3 qui permet de travailler à la fois avec des données discrètes et des données continues. Il permet également de travailler avec des valeurs d'attribut absentes. Enfin, C4.5 élague l'arbre construit afin de supprimer les règles inutiles et de rendre l'arbre plus compact.

C4.5 : Pseudo-code

Le Pseudo-code de l'algorithme C4.5 est :

Algorithme 2 C4.5

Si Tous les échantillons de la liste appartiennent à la même classe **alors**
    Créer simplement un nœud feuille pour l'arbre de décision disant de choisir cette classe
Fin Si
Si Aucune des fonctionnalités ne fournit de gain d'information **alors**
    Créer un nœud de décision plus haut dans l'arbre en utilisant la valeur attendue de la classe.
Fin Si
Si Instance de classe inconnue précédemment rencontrée **alors**
    Créer un nœud de décision plus haut dans l'arbre en utilisant la valeur attendue.
Fin Si
Pour chaque attribut a , trouvez le rapport de gain d'information normalisé à partir du fractionnement sur a .
Soit **a'** l'attribut avec le gain d'information normalisé le plus élevé.
Créez un nœud de décision qui se divise sur **a'** .
Réexaminez les sous-listes obtenues en divisant sur **a'** , et ajoutez ces nœuds en tant qu'enfants de node .

Knn

Présentation

L'algorithme des **K plus proches voisins** ou **K-nearest neighbors (kNN)** est un algorithme de Machine Learning qui appartient à la classe des algorithmes d'apprentissage supervisé simple et facile à mettre en œuvre qui peut être utilisé pour résoudre les problèmes de classification et de régression.

- En classification knn, le résultat est une **classe d'appartenance**.
- En régression kNN, le résultat est la **valeur** pour cet objet. Cette valeur est la moyenne des valeurs des k plus proches voisins.

Knn : Principe

L'intuition derrière l'algorithme des **K plus proches voisins** est l'une des plus simples de tous les algorithmes de Machine Learning supervisé . Il fonctionne de la maniere suivante :

1. Sélectionner le nombre K de voisins
2. Calculer la distance
3. Prendre les K voisins les plus proches selon la distance calculée.
4. Parmi ces K voisins, compter le nombre de points appartenant à chaque catégorie.
5. Attribuer le nouveau point à la catégorie la plus présente parmi ces K voisins.

Les distances pouvant être utilisés pour la classification sont :

Knn :Principe

- **La distance de Minkowski** : avec p un entier positif non nul.

$$d(i,j) = \sqrt[p]{\sum_{r=1}^{M} |x_{ir} - x_{jr}|^p}$$

- **La distance euclidienne :** Elle est un cas particulier de la distance de Minkowski pour p = 2 :

$$d(i,j) = \sqrt{\sum_{r=1}^{M} |x_{ir} - x_{jr}|^2}$$

- **La distance de Manhattan :** Elle est également un cas particulier de la distance de Minkowski pour p = 1

$$d(i,j) = \sum_{r=1}^{M} |x_{ir} - x_{jr}|$$

Knn : Pseudo-code

Le Pseudo-code de l'algorithme knn est :

Algorithme 3 Knn

Entrée(s):

- un ensemble de données D
- une fonction de définition distance d
- Un nombre entier K

Sorties(s): la moyenne ou mode des variables **y** des **K** plus proches observations.

1: **Pour** Une nouvelle observation **X** dont on veut prédire sa variable de sortie **y** **faire**
2: Calculer toutes les distances de cette observation **X** avec les autres observations du jeu de données **D**
3: Retenir les **K** observations du jeu de données **D** les proches de **X** en utilisation le fonction de calcul de distance **d**
4: Prendre les valeurs de **y** des **K** observations retenues :
5: **Si** on effectue une régression **alors**
6: calculer la moyenne (ou la médiane) de **y** retenues
7: **Fin Si**
8: **Si** on effectue une classification **alors**
9: calculer le mode de **y** retenues
10: **Fin Si**
11: **Fin pour**
12: **Retourner** la valeur calculée dans l'étape 3 comme étant la valeur qui a été prédite par K-NN pour l'observation X.

Exercice : Utilisation de l'algorithme KNN

Le tableau suivant contient des données sur des individus d'une population décrite selon deux attributs :attribut 1 et attribut 2. La classe d'un individu peut être : A ou B.

I	A1	A2	C
I1	2	3	A
I2	4	5	B
I3	9	3	A
I4	3	7	B
I5	1	8	B

On veut classer un nouvel individu U ayant comme attributs (1, 4) en utilisant la méthode KNN. Quelle sera la classe de U ?

Bayes naïf

Présentation

La méthode de classification naïve bayésienne ou classifieur bayésien (ou encore estimateur bayésien)est un algorithme d'apprentissage supervisé (supervised machine learning) qui permet de classifier un ensemble d'observations selon des règles déterminées par l'algorithme lui-même. c'est un type de classification probabiliste simple basée sur le théorème de Bayes avec une forte indépendance (dite naïve) des hypothèses. Leur particularité est de prédire la valeur des paramètres du modèle en termes de probabilité.

Bayes naïf

A la base de la classification naïve bayésienne se trouve le théorème de Bayes avec l'hypothèse simplificatrice, dite naïve, d'indépendance entre toutes les paires de variables.
Le théorème de Bayes est donnée par la relation :

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)}$$

Le problème de classification ici revient à estimer la probabilité de chaque classe c_i sachant un vecteur de caractéristiques $\vec{f}$.

Bayes naïf

La probabilité de chaque caractéristique f_j sachant une classe c_i est estimée selon le type de ces valeurs : discrètes, binaires ou continues.

- **Cas des valeurs discrètes** : On utilise la ici loi multinomiale.

  Par exemple, la couleur des cheveux avec les valeurs : brun, auburn, châtain, roux, blond vénitien, blond et blanc. La probabilité d'une caractéristique f_j sachant une classe c_i est le nombre des occurrences de ce critère dans la classe ($|c_i|f_j$) divisé par le nombre de ces occurrences dans tout l'ensemble de données.

$$P(f_j|c_i) = \frac{|c_i|f_j}{\sum_{c_j} |c_j|f_j}$$

Bayes naïf

- **Cas des valeurs binaires** : on utilise la loi de Bernoulli.
- **Cas des valeurs continues** : on utilise la loi normale (loi gaussienne). Par exemple, le poids, le prix, etc. En se basant sur les données d'entrainement avec N échantillons, on calcule l'espérance μ et la variance σ^2 de chaque caractéristique f_j et chaque classe c_i.
  Donc, la probabilité qu'une caractéristique f_j ait une valeur x sachant une classe c_i suit la loi normale.

$$P(f_j = x|c_i) = \frac{1}{2\pi\sigma_{ij}^2} e^{-\frac{(x-\mu_{ij})^2}{2\sigma_{ij}^2}}$$

Bayes naïf

L'espérance μ et la variance σ^2 sont données par :

$$\mu_{ij} = \frac{1}{\sum_{|c_i|} \sum_{k=1}^{|c_i|} x_k / x_k}$$

$$\sigma_{ij}^2 = \frac{1}{\sum_{|c_i|-1} \sum_{k=1}^{|c_i|} (x_k - \mu_{ij})^2 / x_k}$$

Bayes naïf : Exemple

Soit le tableau des données ci-dessous contenant 8 échantillons :

Sexe	Taille (cm)	Poids (kg)	Pointure
masculin	182	81.6	30
masculin	180	86.2	28
masculin	170	77.1	30
masculin	180	74.8	25
féminin	152	45.4	15
féminin	168	68.0	20
féminin	165	59.0	18
féminin	175	68.0	23

On souhaite donc vérifier si un échantillon avec les caractéristiques suivantes : {taille=183, poids=59, pointure=20} est féminin ou masculin.

Bayes naïf : Exemple

Les classes sont : **masculin** et **féminin**.

On a les probabilités suivantes :
P(masculin) $= 4/8 = 0.5$ P(féminin) $= 4/8 = 0.5$

La phase d'apprentissage consiste à calculer l'espérance et la variance de chaque caractéristique et classe.

Sexe	μ **(taille)**	σ^2 **(taille)**	μ **(poids)**	σ^2 **(poids)**	μ **(pointure)**	σ^2 **(pointure)**
masculin	178	29.333	79.92	25.476	28.25	5.5833
féminin	165	92.666	60.1	114.04	19.00	11.333

Bayes naïf : Exemple

On peut à présent déterminer le sexe de l'échantillon avec :

$$f_{i,j}(x) = \frac{1}{2\pi\sigma_{i,j}^2} exp(\frac{-1}{2\sigma_{i,j}^2}(x - \mu_{i,j})^2)$$

pour une variable j dans le groupe i.

Bayes naïf : Exemple

Pour la variable taille (t) dans le groupe masculin (m) on a donc :

$$P(t|m) = f_{t,m}(x) = \frac{1}{2\pi \times 29.333} exp(\frac{-1}{2 \times 29.333}(183 - 178)^2)$$

$$\approx 0,048102$$

En réalisant ce calcul pour chacune des autres caractéristiques on obtient :

$P(\text{taille}|\text{masculin}) = 4.8102 \times 10^{-2}$
$P(\text{poids}|\text{masculin}) = 1.4646 \times 10^{-5}$
$P(\text{pointure}|\text{masculin}) = 3.8052 \times 10^{-4}$

$P(\text{taille}|\text{féminin}) = 7.2146 \times 10^{-3}$
$P(\text{poids}|\text{féminin}) = 3.7160 \times 10^{-3}$
$P(\text{pointure}|\text{féminin}) = 1.1338 \times 10^{-1}$

Bayes naïf : Exemple

Nous rappelons que nous avons obtenu plus haut les probabilités suivantes :

$P(\text{masculin}) = 0.5$

$P(\text{féminin}) = 0.5$

Posons : m=masculin, f=féminin. On aura donc :

$$P_p(\text{m}) = P(\text{m})P(\text{taille}|\text{m})P(\text{poids}|\text{m})P(\text{pointure}|\text{m})$$

$$P_p(\text{f}) = P(\text{f})P(\text{taille}|\text{f})P(\text{poids}|\text{f})P(\text{pointure}|\text{f})$$

Ainsi on a donc :

$P_p(\text{féminin}) = 1.5200 \times 10^{-5}$

$P_p(\text{masculin}) = 1.3404 \times 10^{-10}$

Comme la probabilité (postérieure) féminin est supérieure à la probabilité (postérieure) masculin, l'échantillon est plus probablement de sexe féminin.

Apriori

Présentation des concepts

Présentation

L'algorithme APriori est un algorithme d'exploration de données conçu en 1994, par Rakesh Agrawal et Ramakrishnan Sikrant, dans le domaine de l'apprentissage des règles d'association. Il sert à reconnaitre des propriétés qui reviennent fréquemment dans un ensemble de données et d'en déduire une catégorisation.

Apriori

Présentation des concepts

Définitions

1. Une **règle d'association** est une application de la forme X$\rightarrow$ Y où X et Y sont des ensembles d'items disjoints.
   Dans notre cas , une règle peut s'écrire :

   SI Produit1 **ALORS** Produit2
2. **item** : un élément d'un ensemble (un produit)
3. **itemset** : ensemble de produits (par exemple :{Pain, Lait})
4. **sup(itemset)** : nombre de transactions d'apparition simultanée des produits
5. **card(itemset)** : nombre de produits dans l'ensemble


Apriori

Présentation des concepts

- Une **transaction** représente un ensemble d'articles ou objets dans une base des données
- Nous pouvons représenter les transactions comme :
  1. Liste
  2. Représentation verticale
  3. Représentation horizontale

Apriori

Présentation des concepts

Une Liste

- Chaque ligne représente une transaction
- Chaque ligne liste les items achetés par le consommateur
- Les lignes peuvent avoir un numéro différent de colonnes

Représentation verticale :seulement deux colonnes

- une colonne pour les numéros de la transaction (id)
- Une colonne indiquant un item présent

Représentation horizontale : Les transactions se représentent avec une matrice binaire

- Chaque ligne de la matrice représente une transaction
- Chaque colonne représente un article ou item
- Si un item est présent dans une transaction sera représenté avec un 1
- Si un item est absent sera représenté avec un 0

Exemple

Le tableau ci-dessous contient un ensemble de données relatives aux transactions de vente dans un magasin.

Transactions	Articles
1	{ Yaourt, Tomate, Banane}
2	{ Orange, Pomme }
3	{ Yaourt, Tomate, Orange, Pomme }
4	{ Tomate, Pomme, Banane }
5	{ Yaourt, Tomate,Orange,Banane }

1. Sous quelle forme ces transactions ont été représentées
2. Donner une représentation binaire de ce tableau.

Solution exemple

1. Il s'agit d'une représentation verticale
2. La représentation binaire de ce tableau (représentation horizontale) est :

TID	Yaourt	Tomate	Orange	Pomme	Banane
1	1	1	0	0	1
2	0	0	1	1	0
3	1	1	1	1	0
4	0	1	0	1	1
5	1	1	1	0	1

Apriori

Présentation des concepts

Les règles d'association sont basées sur les mesures suivantes : la **fréquence** , le **support** et la **confiance** .

1. La **fréquence** est le nombre de fois qu'un élément ou une combinaison d'éléments apparaît dans un ensemble de données.
2. Le **support** d'un ensemble d'items fait référence au nombre de transactions (observées) qui le contient. Mathématiquement, le support $\sigma(X)$ d'un ensemble d'items X est défini par : $\sigma(X) = Card(\{t_i | X \subset t_i, t_i \in T\})$ où $Card(A)$ représente le cardinal de l'ensemble A.
3. La **confiance** est la probabilité que deux articles soient vendus ensemble. Elle est définie comme le nombre de fois qu'une combinaison d'articles a été vendue ensemble, divisé par le nombre de fois que le premier article apparaît dans l'ensemble de données.

1. Calcul du support d'une règle d'association :

$$\sigma(X \rightarrow Y) = \frac{\sigma(X \bigcup Y)}{N}$$

N est le nombre de transactions.

2. Calcul de la confiance d'une règle d'association :

$$c(X \rightarrow Y) = \frac{\sigma(X \bigcup Y)}{\sigma(X)}$$

Remarques

- le support est un indicateur de « fiabilité » d'une règle
- La confiance est un indicateur de « précision » d'une règle
- Il existe un autre métrique permettant d'avaluer une règle d'association. Il s'agit de **LIFT** qui est un indicateur de pertinence des règles.

$$Lift(X \rightarrow Y) = \frac{\sigma(X \rightarrow Y)}{\sigma(X) * \sigma(Y)}$$

Apriori

Exercice

Supposons que nous avons un ensemble de données contenant des transactions de vente dans un magasin.

Transactions	Articles
1	{ Pain, Lait, Fromage}
2	{ Pain, Lait, Beurre }
3	{ Pain, Lait, Beurre, Fromage }
4	{ Pain, Lait, Beurre }
5	{ Pain, Lait, Fromage }

1. Combien d'items possibles peut-on former à partir de ce tableau des transactions ?
2. Que représente un item ? Donner un exemple.
3. Considérons la règle suivante : {Pain, Lait}→ {Fromage}. Déterminer son support et sa confiance et son Lift.

Apriori :Principe

L'algorithme Apriori s'execute en deux étapes :

1. Génération de tous les itemsets fréquents c'est-à-dire :

$$\text{IF} = \{ X_i \subseteq T | \text{ supp } (X_i) = X_i.count \geqslant \text{minsupp}, \quad \text{i=1,2..,n } \}$$

2. Génération de toutes les règles d'associations de confiance à partir des itemsets fréquents, c'est-à-dire

$$\{ X_i, Y_j \subseteq IF | X_i \cap Y_j = \varnothing \wedge \text{Conf}(X_i \rightarrow Y_j) \geqslant \text{minconf} \quad \text{i=1,2,...,p j=1,2,...,q } \}$$

minsupp est l'indice de support minimum donné, et **minconf** l'indice de confiance donné.

Apriori

Le Pseudo-code de l'algorithme Apriori est :

Algorithme 4 Apriori

Entrée(s): Base de données de transactions D, Seuil de support minimum σ ;

Sorties(s): Ensemble des items fréquents

1: i←1
2: $C_1 \leftarrow$ ensemble des motifs de taille 1 (un seul item)
3: **Tant que** $C_i \neq \emptyset$ **faire**
4: Calculer le Support de chaque motif m $\in C_i$ dans la base
5: $F_i \leftarrow \{m \in C_i | \text{ support}(m) \geqslant \sigma \}$
6: $C_{i+1} \leftarrow$ toutes les combinaisons possibles des motifs de F_i de taille i + 1
7: i← i + 1
8: **Fin Tant que**
9: **Retourner** $\bigcup_{(i \geqslant 1)} F_i$

Apriori :Exemple

Le tableau ci-dessous représente le contenu du panier d'une ménagère.

TID	Items
1	{ Pain, Lait}
2	{ Pain, Couches,Bière, Oeufs }
3	{ Lait, Couches, Bière, Djino }
4	{ Pain, Lait, Couches, Bière }
5	{ Pain, Lait, Couches, Djino }

Pour découvrir les relations cachées dans cette base de données, nous allons effectuer l'analyse des associations.

Apriori :Exemple

La représentation binaire des données du tableau précédent est donnée ci-dessous :

TID	Pain	Lait	Couches	Bière	Oeufs	Djino
1	1	1	0	0	0	0
2	1	0	1	1	1	0
3	0	1	1	1	0	1
4	1	1	1	1	0	0
5	1	1	1	0	0	1

Apriori :Exemple

Par exemple pour la règle {Lait, Couches} $\rightarrow$ {Bière}. Le support de l'ensemble {Bière, Couches, Lait} étant 2 et le nombre total de transactions étant 5, le support de la règle est donc : $\frac{2}{5}$=0,4

Sa confiance = $\frac{support\{Bière, Couches, Lait\}}{support\{Lait, Couches\}} = \frac{2}{3} = 0{,}67$ car on a 3 transactions contenant {Lait, Couches}.

■ **Recherche de règles d'associations en utilisant l'algorithme Apriori**

Fixons un degré d'exigence sur les règles à extraire. Par exemple :
Support min : 2 transactions et **Confiance min** =75%
L'idée est surtout de contrôler (limiter) le nombre de règles produites.
Nous allons procéder ici en deux étapes :

Apriori :Exemple

Étape 1 : Génération des ensembles d'items fréquents (support $\geq$ support min.)

Pour le faire, nous allons utiliser le graphe pour énumérer tous les ensembles d'items possibles.

Sur ce graphe ci-dessous :

P= Pain

L = Lait

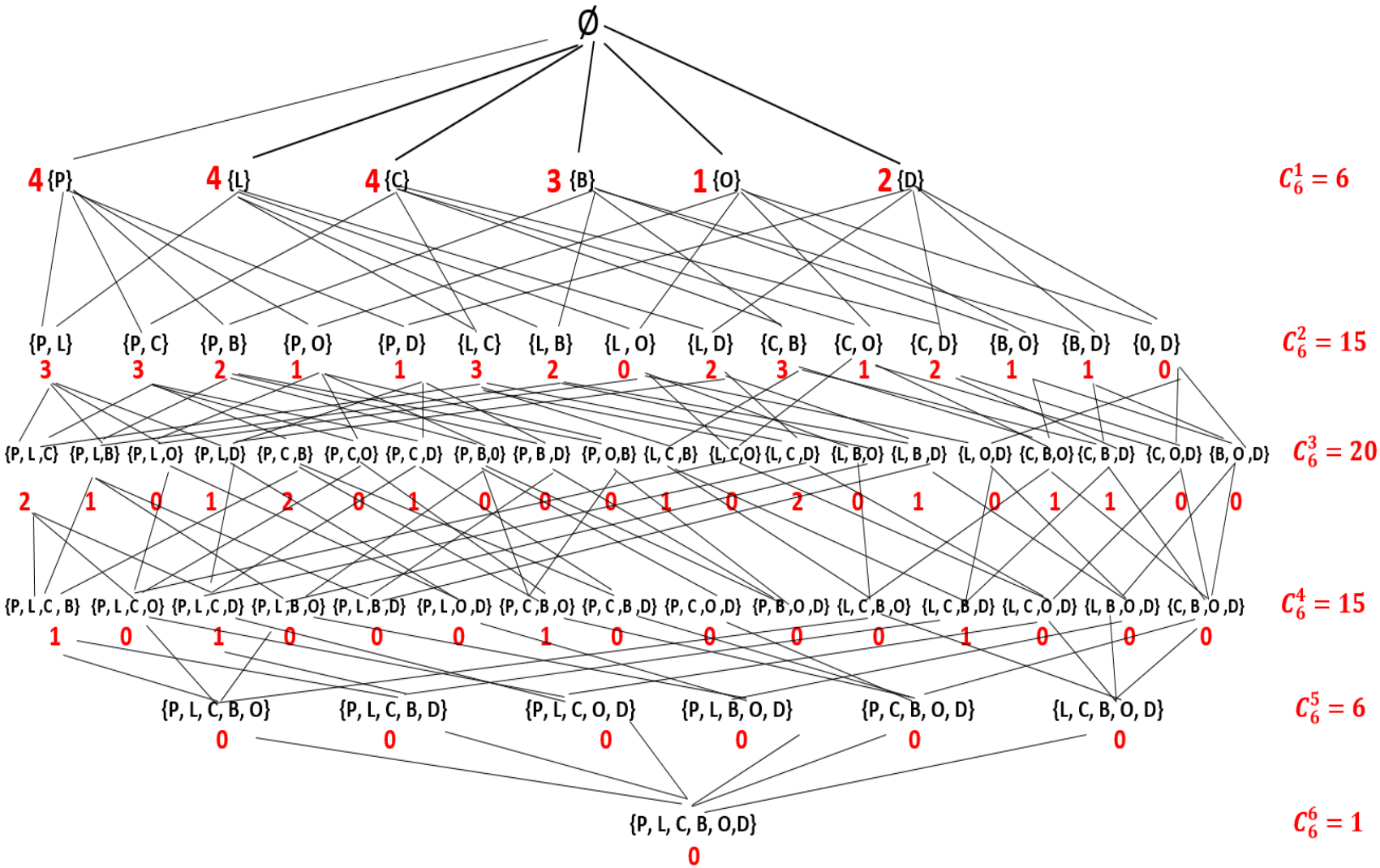
C =Couches

B = Biere

O = Oeufs

D = Djino

Apriori :Exemple



Apriori :Exemple

Le support minimal étant de 2, nous retenons donc de ce graphe, tous les itemsets fréquents c'est-à-dire les itemsets donc le support est supérieur ou égal à 2.
Le tableau ci-dessous, donne la liste de ces itemsets fréquents :

Item	{P}	{L}	{C}	{B}	{D}	{P,L}	{P,C}	{P,B}	{L,C}	{L,B}	{L,D}	{C,B}	{C,D}	{P,L,C}	{P,C,B}	{L,C,D}
Sup	4	4	4	3	2	3	3	2	3	2	2	3	2	2	2	2

Apriori :Exemple

Étape 2 : Génération des règles à grande confiance à partir des items fréquents précédents trouvés précédemment : Règles fortes (conf. $\geq$ conf. min.)

Il s'agit ici de déduire les règles à partir des itemsets fréquents et limiter par la suite la prolifération des règles en utilisant le critère **confiance min**=0,75.

Nous utiliserons par conséquent les itemsets fréquents avec au moins deux éléments et nous allons tester toutes les combinaisons possibles :

Itemset	{P,L}	{P,C}	{P,B}	{L,C}	{L,B}	{L,D}	{C,B}	{C,D}	{P,L,C}	{P,C,B}	{L,C,D}
Support	3	3	2	3	2	2	3	2	2	2	2

Apriori :Exemple

Calculons donc la confiance de chaque règle :

{ P, L }	$P \longrightarrow L$: conf = 3 / 4 = 75% **(accepté)**
	$L \longrightarrow P$: conf = 3 / 4 = 75% **(accepté)**
{ P, C }	$P \longrightarrow C$: conf = 3 / 4 = 75% **(accepté)**
	$C \longrightarrow P$: conf = 3 / 4 = 75% **(accepté)**
{ P, B }	$P \longrightarrow B$: conf = 2 / 4 = 50% **(refusé)**
	$B \longrightarrow P$: conf = 2 / 3 = 66,6% **(refusé)**
{ L, C }	$L \longrightarrow C$: conf = 3 / 4 = 75% **(accepté)**
	$C \longrightarrow L$: conf = 3 / 4 = 75% **(accepté)**

Apriori :Exemple

{ L, B }	L ⟶ B : conf = 2 / 4 = 50% **(refusé)**
	B ⟶ L : conf = 2 / 3 = 66,6% **(refusé)**
{ L, D }	L ⟶ D : conf = 2 / 4 = 50% **(refusé)**
	D ⟶ L : conf = 2 / 2 = 100% **(accepté)**
{ C, B }	C ⟶ B : conf = 3 / 4 = 75% **(accepté)**
	B ⟶ C : conf = 3 / 3 = 100% **(accepté)**
{ C, D }	C ⟶ D : conf = 1 / 4 = 25% **(refusé)**
	D ⟶ C : conf = 1 / 2 = 50% **(refusé)**

Apriori :Exemple

{ P,L, C }

$P \longrightarrow \{L,C\}$: conf = 2 / 4 = 50% **(refusé)**

$\{L,C\} \longrightarrow P$: conf = 2 / 3 = 66,6% **(refusé)**

{ P,C, B }

$P \longrightarrow \{C,B\}$: conf = 2 / 4 = 50% **(refusé)**

$\{C,B\} \longrightarrow P$: conf = 2 / 3 = 66,6% **(refusé)**

{ L,C, D }

$L \longrightarrow \{C,D\}$: conf = 2 / 4 = 50% **(refusé)**

$\{C,D\} \longrightarrow L$: conf = 2 / 2 = 100% **(accepté)**

Toutes les règles ayant une confiance supérieure ou égale à 75% sont acceptées.

Apriori :Limites

Malgré sa simplicité, l'algorithme Apriori présente quelques limites :

- Il n'est pas envisageable de chercher toutes les règles d'associations pour ensuite sélectionner celles qui ont un support et une confiance suffisants.Cela est très couteux à gérer un grand nombre d'ensembles candidates. Par exemple pour un ensemble de d items, le nombre total de règles possibles est de $R = 3^d - 2^{d+1} + 1$. Si d =6, on R=602.
- Il es fastidieux de de numériser plusieurs fois sur la base des données et vérifier un grand nombre de candidats par correspondance de motif,ce qui est particulièrement vrai pour l'exploitation des longs motifs.

Afin de surmonter les inconvénients rencontrés dans l'algorithme **Apriori**, l'on utilisera l'algorithme **FP-Growth**

Exercice

Exercice

Vous disposez d'un ensemble de données représentant les transactions des clients dans un magasin de vêtements. Chaque transaction se compose d'articles achetés par un client.
Transaction 1 : {Chemise, Pantalon, Cravate}
Transaction 2 : {Chemise, Veste, Chaussures}
Transaction 3 : {Pantalon, Chaussures, Chapeau}
Transaction 4 : {Chemise, Pantalon, Chaussures}
Transaction 5 : {Veste, Chapeau}

1. Donner une représentation binaire de ces données
2. Utilisez l'algorithme Apriori pour trouver tous les ensembles fréquents avec un support minimum de 2 transactions.
3. En utilisant les ensembles fréquents trouvés à l'étape précédente, générez toutes les règles d'association possibles avec une confiance minimale de 50%.

Exercice

Exercice

4. Calculez le support et la confiance pour chaque règle d'association générée.
5. Identifiez les règles d'association intéressantes qui satisfont à la fois le support minimum et la confiance minimale.
6. Interprétez les règles d'association intéressantes trouvées en termes de comportement d'achat des clients dans le supermarché.

Présentation

- K-means est un algorithme non supervisé de clustering . Il permet de regrouper en **K clusters** distincts les observations du data set(ensemble des données).
- Les données similaires se retrouveront dans un même cluster. Par ailleurs, une observation ne peut se retrouver que dans un cluster à la fois (exclusivité d'appartenance). Une même observation, ne pourra donc, appartenir à deux clusters différents.
- Pour pouvoir regrouper un jeu de données en K cluster distincts, l'algorithme K-Means a besoin d'un moyen de comparer le degré de similarité entre les différentes observations.
- Ainsi, deux données qui se ressemblent, auront une distance de dissimilarité réduite, alors que deux objets différents auront une distance de séparation plus grande.

K-means

Application

Les champs d'application de K-Means sont nombreux, il est notamment utilisé en :

- la segmentation de la clientèle en fonction d'un certain critère (démographique, habitude d'achat etc. ...)
- Utilisation du clustering en Data Mining lors de l'exploration de données pour déceler des individus similaires. Généralement, une fois ces populations détectées, d'autres techniques peuvent être employées en fonction du besoin.
- Clustering de documents (regroupement de documents en fonction de leurs contenus. Pensez à comment Google Actualités regroupe des documents par thématiques.)

K-means

Fonctionnement

Le K-means divise les données en k clusters en minimisant la somme des distances au carré entre chaque point et le centroïde de son cluster. Il alterne entre l'affectation des points aux centroïdes les plus proches et la mise à jour des positions des centroïdes jusqu'à convergence.

1. Sélectionner aléatoirement k centroïdes initiaux.
2. Répéter jusqu'à convergence :
   - Assigner chaque point au centroïde le plus proche.
   - Mettre à jour les positions des centroïdes en calculant les moyennes des points assignés.

K-means

Pseudo-code

Ci-dessous le Pseudo-code de l'algorithme K-means

1. **Entrées :**
   - K : le nombre de cluster à former
   - training set : Ensemble ou matrices des données
2. choisir aléatoirement le k points qui seront considérés comme les centroides (les centres de clusters)
3. **Répéter** :
   - (Ré)affecter chaque objet O au cluster C_i de centre M_i tel que dist(O,M_i) est minimal
   - Recalculer M_i de chaque cluster (le barycentre)
4. **jusqu**'à la convergence

K-means I

Exemple

Soit A={1,2,3,6,7,8,13,15,17} un ensemble des données numériques. Créer 3 clusters à partir de A

- On prend 3 objets au hasard. Supposons que c'est 1, 2 et 3. Ca donne C1={1}, M1=1, C2={2}, M2=2, C3={3} et M3=3
- Chaque objet O est affecté au cluster au milieu duquel, O est le plus proche. 6 est affecté à C3 car dist(M3,6)<dist(M2,6) et dist(M3,6)<dist(M1,6) On a :
  C1={1}, M1=1
  C2={2}, M2=2
  C3={3, 6,7,8,13,15,17}, M3=69/7=9.86
- dist(3,M2)<dist(3,M3)→ 3 passe dans C2. Tous les autres objets ne bougent pas. C1={1}, M1=1, C2={2,3}, M2=2.5,C3={6,7,8,13,15,17} et M3= 66/6=11

K-means II

Exemple

- dist(6,M2)<dist(6,M3) $\rightarrow$ 6 passe dans C2. Tous les autres objets ne bougent pas. C1=$\{1\}$, M1=1, C2=$\{2,3,6\}$, M2=11/3=3.67, C3=$\{7,8,13,15,17\}$, M3= 12
- dist(2,M1)<dist(2,M2) $\rightarrow$ 2 passe en C1. dist(7,M2)<dist(7,M3)$\rightarrow$ 7 passe en C2. Les autres ne bougent pas. C1=$\{1,2\}$, M1=1.5, C2=$\{3,6,7\}$, M2=5.34, C3= $\{8,13,15,17\}$, M3=13.25
- dist(3,M1)<dist(3,M2)$\rightarrow$ 3 passe en 1.
- dist(8,M2)<dist(8,M3)$\rightarrow$ 8 passe en 2 C1=$\{1,2,3\}$, M1=2, C2=$\{6,7,8\}$, M2=7, C3=$\{13,15,17\}$, M3=15
- Plus rien ne bouge, on s'arrete car l'algorithme converge. On a donc les classes finales suivantes : **C1=$\{1,2,3\}$**, **C2=$\{6,7,8\}$** et **C3=$\{13,15,17\}$**

K-means

Limites

Quelques limites de cet algorithme sont :

- Un nombre K grand peut conduire à un partitionnement trop fragmenté des données. Ce qui empêchera de découvrir des patterns intéressants dans les données. Par contre, un nombre de clusters trop petit, conduira à avoir, potentiellement, des cluster trop généralistes contenant beaucoup de données. Dans ce cas, on n'aura pas de patterns "fins" à découvrir.
- Pour un même jeu de données, il n'existe pas un unique clustering possible. La difficulté résidera donc à choisir un nombre de cluster K qui permettra de mettre en lumière des patterns intéressants entre les données. Malheureusement il n'existe pas de procédé automatisé pour trouver le bon nombre de clusters.

Exercice : Utilisation de l'algorithme Kmeans

Soit l'ensemble D des entiers suivants : D= { 2, 5, 8, 10, 11, 18, 20 } . On veut répartir les données de D en trois (3) clusters, en utilisant l'algorithme Kmeans. La distance d entre deux nombres a et b est calculée ainsi : d(a , b) = |a - b| (la valeur absolue de a moins b)

1. Appliquez l'algorithme Kmeans en choisissant comme centres initiaux des 3 clusters respectivement : 8, 10 et 11. Montrez toutes les étapes de votre calcul.
2. Donnez le résultat final et précisez le nombre d'itérations qui ont été nécessaires.
3. Peut-on avoir un nombre d'itérations inférieur pour ce problème ? Discutez.

Beaucoup d'autres algorithmes de web mining existent, nous avons par exemple les agorithmes suivants :

- One Rule,
- Co-training,
- CART
- SLIQ,
- AdaBoost,
- Learn++,
- Eclat,
- SSDM,
- kDCI,
- GSP,
- SPADE,
- ...

CHAPITRE 3

Le Web Content Mining

Introduction

- Le Web contient une masse importante des documents pouvant être structurés , semi-structurés ou non structurés.
  1. Les **données structurées** : représentent des données sous forme de base de données plus facile à extraire et à manipuler par rapport aux textes non structurés.
  2. Les **données semi-structurées** : représentent des données sous forme de documents XML contenant des informations
  3. Les **données non structurées** : représentent des données qui ne sont pas organisées dans un format facile à traiter (texte, son, image)
- analyser ces données peut fournir un grand nombre d'informations qui peuvent se révéler utiles pour un décideur.
- le Web Content Mining traite donc de l'analyse de ces contenus du Web.

Définition

Définiton

- Le web Contenu Mining est le domaine qui fournit des méthodes permettant l'extraction, l'organisation, la gestion et la découverte automatique de la quantité énorme d'informations et de ressources disponibles dans le web.
- Le Web Content Mining peut être défini comme le processus d'extraction des informations à partir de différentes sources de données dans le web.

Les données textuelles représentent les données les plus répandues sur le Web et constituent également une source d'information qui permettrait d'extraire de la connaissance en faisant appel aux techniques de fouille des données. La branche du Web Content Mining permettant d'analyser les données textuelles est le **Text Mining**


Tâches de fouille du texte

- **L'extraction d'information** qui consiste à identifier et rechercher les mots-clés et les phrases dans le texte.
- La **visualisation de l'information** qui utilise l'extraction des figures et l'indexation des termes clés qui construisent une représentation graphique.
- La **catégorisation** qui consiste à identifier les principaux thèmes et à incrémenter le nombre de mots dans ce document permettant ainsi le classement de la page.
- Le **clustering** qui consiste à regrouper les documents similaires ce qui permet à l'utilisateur de sélectionner le thème qui l'intéresse
- **indexation automatique** pour faciliter la recherche de la page par les outils de recherche(moteurs de recherche, annuaire de recherche,...)


Collecte des données du Web

- L'étape de collecte des données est la première étape du processus de data mining. Elle consiste à recueillir les données brutes à partir de diverses sources, telles que des fichiers texte, des bases de données, des fichiers Excel, des fichiers de log, des flux de données en direct, etc.
- Le succès de l'ensemble du processus de data mining dépend en grande partie de la qualité et de la quantité des données collectées. Par conséquent, il est important de veiller à ce que les données collectées soient pertinentes et complètes, car cela garantira que l'analyse des données est précise et pertinente pour l'objectif visé.

Collecte des données du Web

Le Web est une source incontournable d'informations. Pour accéder à ces informations, on pourra utiliser plusieurs méthodes.Ces méthodes se déclinent en trois variantes :

- L'accès direct (type surfer sur Web) ;
- L'accès via la couche des moteurs de recherche
- L'accès via la couche des méta-moteurs

Dans tous les cas, l'information collectée est réduite à des **URLs** ou à des documents **HTML** qui sont structurés en des balises permettant d'inserer plusieurs autres types de données (texte, images, vidéos, sons, liens, ...). En fonction de type de la donnée recherchée, on s'intéressera à la balise appropriée.


Collecte des données du Web

Sources des données

Voici les principales sources de données utilisées pour le web mining :

1. **Les fichiers de données stockés sur les systèmes de stockage en ligne** : Les entreprises stockent souvent des données importantes dans des fichiers stockés sur des systèmes de stockage tels que les systèmes de fichiers distribués, les systèmes de stockage cloud et les serveurs. Ces fichiers contiennent souvent des informations précieuses telles que des données de vente, des données de transactions, des données de suivi de l'inventaire, etc.
2. **Les bases de données relationnelles** : Les bases de données relationnelles stockent des données dans des tables organisées en colonnes et en lignes. Les bases de données peuvent contenir des informations sur les clients, les ventes, les stocks, les employés, les fournisseurs, etc.

3. **Les fichiers log** : Les fichiers log sont des fichiers générés par les applications et les systèmes d'exploitation pour enregistrer les événements tels que les erreurs, les avertissements et les informations sur l'utilisation. Les fichiers log contiennent souvent des informations précieuses sur les performances du système, les erreurs et les activités des utilisateurs.
4. **Les flux de données en direct** : Les flux de données en direct sont des données qui sont générées en temps réel à partir de diverses sources telles que les capteurs, les caméras, les enregistreurs de données, etc. Les flux de données en direct peuvent fournir des informations précieuses sur les tendances en temps réel, les anomalies et les modèles.

5. **Les données provenant de sources tierces** : Les données provenant de sources tierces, telles que les réseaux sociaux, les pages web, les sources d'informations publiques, les données environnementales, les données démographiques, etc., peuvent fournir des informations précieuses pour le data mining.

Remarques

1. Une fois les données collectées, il est **important de veiller à ce qu'elles soient stockées dans un format qui facilite leur traitement et leur analyse**. Cela peut inclure la conversion de fichiers de données en un format standard tel que CSV (Comma Separated Values), la normalisation des données et la suppression des données redondantes ou non pertinentes.
2. En suivant ces bonnes pratiques, vous serez en mesure de collecter des données de qualité et de les stocker de manière appropriée pour le processus de data mining.

Important !

- Il est également important de noter que les données collectées doivent être conformes aux lois et réglementations en matière de protection des données personnelles. Les entreprises doivent se conformer à des normes strictes pour assurer la confidentialité et la sécurité des données collectées, afin d'éviter tout risque de violation de la vie privée des individus.
- Il est également essentiel de définir les objectifs de l'analyse des données dès le début du processus de collecte de données. Cela permet de déterminer quelles données sont nécessaires pour atteindre les objectifs et quelles sont les sources de données appropriées pour les obtenir

Collecte des données du Web

Outils de collecte des données

Il existe plusieurs outils de collecte de données pour collecter et extraire des données à partir de diverses sources. Voici quelques-uns des outils les plus couramment utilisés :

- **Google Forms** : Google Forms est un outil de collecte de données gratuit qui permet de créer des formulaires personnalisés pour collecter des données auprès des utilisateurs. Les données collectées peuvent être exportées dans différents formats, notamment CSV et Excel.
- **SurveyMonkey** : SurveyMonkey est un outil de collecte de données en ligne qui permet de créer des sondages et des enquêtes en ligne pour collecter des données auprès des participants. Les résultats peuvent être exportés dans différents formats, y compris CSV et Excel.

- Les **navigateurs** gratuites qui permet de collecter des données à partir de sites web en utilisant des techniques de web scraping. L'outil peut extraire des données à partir de pages web, de tableaux et de fichiers PDF.
- **Scrapy** : Scrapy est un framework open source de web scraping en Python. Il permet de collecter des données à partir de sites web en utilisant des scripts Python personnalisés pour extraire les données souhaitées.

Définition

Le web scraping (parfois appelé harvesting) est une technique d'extraction du contenu de sites Web, via un script ou un programme, dans le but de le transformer pour permettre son utilisation dans un autre contexte.

Pourquoi le web Scraping pour un étudiant ?

Acquérir des données textuelles pour une recherche, un mémoire, pour organiser un projet (adresses mail, numéros de téléphone, liste d'articles) et/ou une veille informationnelle, pour archiver...


Collecte des données du Web

Le Web Scraping

1. Utiliser des « robots » pour « aspirer » des sites web
2. Un **web scraper** est un programme informatique qui lit le code html des pages web pour extraire des données qui sont présentées sur des sites web.
3. Scraper n'est pas une activité illégale en soi. Mais une utilisation des données webscrapées peut aller contre l'éthique ou être véritablement illégale... (**Rappel** : toujours lire et respecter les conditions générales d'utilisation (CGU) et les conditions générales de vente (CGV) avant de web scrapper).
4. De nombreux outils disponibles parmi lesquels : Import.io, Scrapy, Outwit Hub, Gromoteur
5. Webscraper est une extension disponible sous Google Chrome qui permet d'extraire les données d'un site internet très rapidement. L'extension de Google Chrome : http ://webscraper.io/

Outil Python : Scrapy(scrapy.org) :

- Python-based scraping and crawling framework
- More IT oriented : coding skills required
- Open source
- Large user community
- Used by some NSI's for various scraping tasks

Outil Python : JSOUP

- bibliothèque java qui est un outil très important à utiliser dans la collecte des données se trouvant sur une page Web.
- d'analyser le HTML à partir de son adresse URL et d'extraire ses contenus.
- Manipuler les éléments HTML , les attributs et les textes
- Trouver et extraire les données en utilisant les sélecteurs CSS

Pré-traitement des données textuelles

Avant de pouvoir analyser les données textuelles , nous devions impérativement les transformées en entrées valides compréhensibles par les différents algorithmes. les opérations de pré-traitements classiques de textes sont donc :

- **Normalisation** : Elle consiste de mettre le texte à la même casse, souvent tout en minuscule.
- **Tokenisation** Elle consiste à segmenter un document texte en tokens de mots, séquences de mots ou carrément de phrases, mais généralement elle s'opère souvent sur des mots.
  Un **token** est une unité définie comme une séquence de caractères comprise entre deux séparateurs ; les séparateurs étant les blancs, les signes de ponctuation et certains autres caractères comme les guillemets ou les parenthèses.

Pré-traitement des données textuelles

- **Élimination des mots vides**
  Certains de tokens sont présents dans tous les textes du corpus, c'est ce que nous appelons les mots vides : les articles, les prépositions, les déterminants, les adverbes... comme "la ,le , dans, car," dans la langue française, et "the, and, after" dans la langue anglaise. Ils représentent 30% des mots dans un texte. La présence de ces mots n'apporte absolument aucune différence tant sur le plan sémantique que sur le plan lexical. Cela veut dire que leur présence dans tous les textes du corpus les rend non discriminants et du coup leurs utilisation pour une tâche de classification s'avère inutile. Par contre, leur suppression réduit la dimension de notre document vecteur, par conséquent, le temps de traitement et le temps d'apprentissage seront réduit considérablement

Pré-traitement des données textuelles

- **La racinisation (Stemming)**
  Lors de certaines représentations vectorielles, chaque mot présent dans le corpus est considéré comme un descripteur. Cette façon de faire contient certaines irrégularités, notamment pour les verbes à l'infinitif et les verbes conjugués (développer, développe, développé, développèrent...), nous remarquons que ces derniers ont le même sens mais chacun est considéré comme un descripteur à part entière. La racinisation vient parer à ce problème, en considérant uniquement la racine de ces mots plutôt que les mots en entier sans se soucier de l'analyse grammaticale.

Pré-traitement des données textuelles

- **Lemmatisation**
  Elle met en évidence l'analyse grammaticale, elle ramène les termes à leur forme canonique en mettant tous les noms au singulier, les adjectifs au masculin singulier, et tous les verbes conjugués à l'infinitif.
  Cette technique présente certaines ambigüités dans le cas ou un descripteur représente un même mot qui a deux sens différents, notamment dans l'exemple suivant : "actions" et "action" seront représentés par leur forme singulière par le descripteur "action", cependant, le mot peut avoir deux sens bien différents selon son utilisation ; En économie le mot "actions" peut signifier les actions en bourse, qui n'a rien avoir avec "un film d'action".

Pondérations ou calcul des fréquences

Frequence d'un terme

Elle désigne le nombre de fois qu'un certain descripteur est apparu dans chacun des documents d'un corpus.
À partir de ces fréquences, que nous nommons communément "poids", nous pouvons dire qu'un descripteur quelconque est discriminant ou pas, par rapport à un document donné, si son poids est élevé ou pas, respectivement

Il existe plusieurs manières de définir un ensemble de descripteurs :

- Le sac à mots (bag of words)
- Les N-grams
- Fréquence des termes (TF)
- Fréquence documents inverses (IDF)
- TFIDF

Pondérations ou calcul des fréquences

Le sac à mots (bag of words)

- La façon la plus simple et la plus évidente pour la représentation d'un document texte par un document vecteur, est d'utiliser les mots comme descripteurs
- Cette technique conserve le sens naturel des descripteurs.
- Les mots sont regroupés en vrac et traités d'une manière indépendante, ce qui nuit considérablement à la sémantique du texte.

Pondérations ou calcul des fréquences

Le sac à mots (bag of words)

Le gouvernement Camerounais s'est engagé dans un programme de développement de l'enseignement des TIC à travers la généralisation de l'enseignement de l'informatique à tous les niveaux. La mise en œuvre de cette détermination dans le secteur de l'éducation s'exprime à travers des actes administratifs et réglementaires dont le plus important est : L'arrêté N° 3745/D/63/ MINEDUC/CAB du 17/06/2003 portant introduction de l'Informatique dans les programmes de formation des 1er et 2nd Cycles de l'enseignement secondaire général et des ENIEG, rappelons que ces cours étaient déjà dispensés en enseignement technique (EST) et l'entrée en vigueur des programmes d'enseignement dès l'année scolaire 2003/2004.

Mot	Occurr.	Mot	Occurr.	Mot	Occurr.
à	2	en	3	MINEDUC	1
actes	1	engagé	1	mise	1
administratifs	1	ENIEG	1	niveaux	1
arrêté	1	enseignement	5	œuvre	1
CAB	1	entrée	1	plus	1
camerounais	1	est	2	portant	1
cette	1	et	4	programme	3
cycles	1	étaient	1	rappelons	1
dans	3	exprime	1	réglementaires	1
de	8	formation	1	scolaire	1
déjà	1	général	1	secondaire	1
des	5	généralisation	1	secteur	1
dès	1	gouvernement	1	technique	1
détermination	1	important	1	TIC	1
développement	1	informatique	2	tous	1
dispensés	1	introduction	1	travers	1
dont	1	la	2	un	1
du	1	le	3	vigueur	1
éducation	1	les	2		

Pondérations ou calcul des fréquences

Les N-grams

N-grams est une méthode de représentation de documents texte qui consiste à partager ce dernier en séquences de *n* caractères.
Prenons l'exemple de la phrase "**La classification supervisée** " et essayons de la représenter en ngrams caractères.

- si n=2 nous aurons : "La", "a ", " c", "cl", "la", "as", "ss", "si", "if", etc.
- si n=3 nous aurons : "La ","a c", " cl", "cla", "las", "ass", "ssi", "sif", "ifi", etc.
- si n=4 nous aurons : "Lac ", "a cl", " cla", "clas", "lass", "assi", "ssif", "sifi", "ific", etc

Pondérations ou calcul des fréquences

Fréquence des termes (TF)

Nous dénombrons plusieurs manières de calcul de la TF :

- **TF absolue** : c'est le nombre de fois qu'un terme apparait dans un texte donné.

$$\mathbf{TF = NT}$$

(où NT est le nombre de fois où le terme est apparu dans le texte.)

- **TF relative** : C'est le rapport entre le nombre de fois qu'un terme est apparu dans le texte sur le nombre de tous les termes du texte.

$$\mathbf{TF = \frac{NT}{ST}}$$

- **TF booléenne** : se contente juste de la présence ou de l'absence du terme dans le texte.

$$\mathbf{TF = 0 \text{ ou } 1}$$

Pondérations ou calcul des fréquences

Fréquence des termes (TF)

- **Fréquence documents inverses (IDF)** : Elle mesure en quelque sorte le degré de rareté d'un terme, non pas dans un document, mais dans tous les documents d'un corpus. Elle est définie par cette équation :

$$\mathbf{IDF} = \mathbf{log}\left(\frac{\mathbf{N_{doc}}}{\mathbf{Doc_T}}\right)$$

Où N_{doc} est le nombre de documents dans le corpus, et Doc_T est le nombre de documents dans lesquels le terme est apparu. Si le terme est très présent dans tout le corpus alors le rapport sera égal à 1 et $\mathbf{IDF = 0}$ donc le terme est neutralisé. Si par contre il apparait dans un seul document la valeur est maximale.

$$\mathbf{IDF} = \mathbf{log}(\mathbf{N_{doc}})$$

Pondérations ou calcul des fréquences

Fréquence des termes (TF)

- **TFIDF** : Elle est une combinaison de l'abondance particulière et de la rareté générale d'un terme dans un corpus.Elle est calculée avec la formule suivante :

$$\mathbf{TFIDF = TF * log\ (\frac{N_{doc}}{Doc_T})}$$

où **TF** est relative ou absolue.

Étiquetage du document

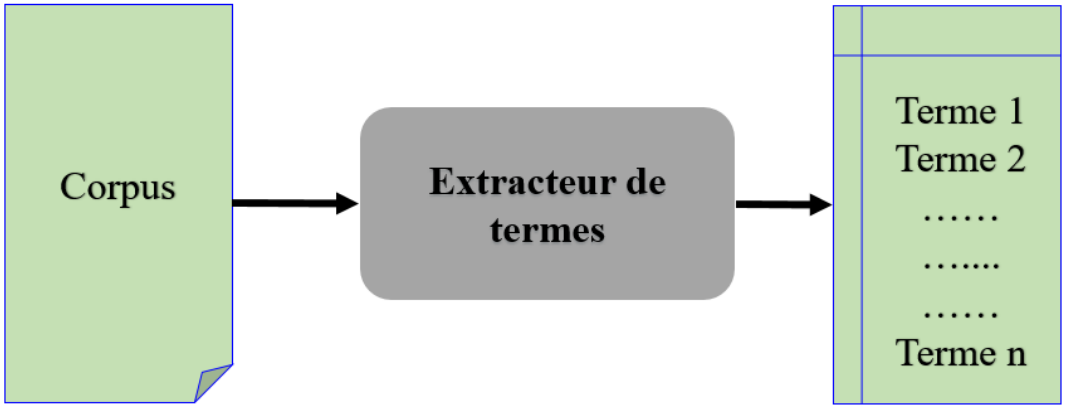
- L'étiqueteur réalise l'analyse morpho-syntaxique qui est une étape qui peut être considérée comme préliminaire à tout traitement linguistique plus poussé sur un texte, notamment l'analyse syntaxique.
- Elle consiste à affecter des étiquettes morpho-syntaxiques propres à chaque mot d'une phrase d'un texte (catégorie grammaticale, informations morphologiques comme le genre, le nombre...).

Par exemple l'étiquetage correct de la phrase "**Isaac a mangé une pomme**" est :

Isaac	a	mangé	une	pomme
Nom	verbe	verbe	déterminant	Nom

Extraction des termes

L'extraction des termes est une étape du processus de classification automatique qui consiste à analyser un corpus et proposer à l'utilisateur les termes qui s'y trouvent.



L'identification d'un terme repose sur le lien qu'on peut établir entre son sens et un domaine de spécialité, donc sur des connaissances extra-linguistiques. Par exemple, après avoir soumis un texte d'informatique à un extracteur de termes, on pourra avoir des termes comme : ordinateur, logiciel, programme, page web, internet, site web, navigateur, tablette, écran, souris,...

Classification d'un document

- **Classer un document** c'est tout simplement mettre une étiquette sur son contenu , lui faisant ainsi appartenir à un groupe ou classe bien définie.
- La classification (ou catégorisation) d'un document est l'une des tâches de traitement du langage naturel (NLP) les plus courantes.
- Elle joue un rôle essentiel dans de nombreuses tâches de gestion et de récupération de l'information.
- Sur le Web, la classification des pages Web est essentielle à l'exploration ciblée, au développement assisté d'annuaires Web, à l'analyse de liens Web spécifiques à un sujet, à la publicité contextuelle et à l'analyse de la structure du Web.


Classification des documents

Définition

Définition

Un classificateur d'un document est une fonction booléenne (f) qui associe automatiquement un document (D) à la classe (C) :

$$f : D \rightarrow C$$

Classification des documents

Formes de classification

- **Classification Binaire :** Il s'agit tout simplement d'une classification à 2 classes comme l'illustre la figure 1.1. Par exemple, un système de détection de SPAM classifie un e-mail comme étant soit un "SPAM", soit un "NON-SPAM".
- **Classification Multi-Classe :** Elle consiste à associer un document à une classe parmi plusieurs.
- **Classification Multi-Label :** Elle consiste à associer le texte en entrée à une ou plusieurs classes.
- **Classification en Cascade :** Il s'agit d'un classifieur composé de plusieurs classifieurs mis l'un à la suite de l'autre. Le but étant de classifier suivant des sous-classes.


Classification des documents

Types de classification

La classification automatique d'un texte peut se faire de deux façons :

1. **La classification non supervisée (clustering)** : Elle consiste à apprendre à classer sans supervision. Au début du processus on ne dispose ni de la définition des classes, ni du nombre. C'est l'algorithme de classification qui va déterminer ces informations.
2. **La classification supervisée (catégorisation)** : Contrairement à l'apprentissage non supervisé, on commence ici par un ensemble de classes connues et définies à l'avance. on dispose aussi d'une sélection initiale de données dont la classification est connue. Ces données sont supposées indépendantes et identiquement distribuées. Elles nous servent pour l'apprentissage de l'algorithme. L 'algorithme réalise donc la classification selon le modèle qu'il a appris.

Classification des documents

Évaluation d'un classificateur

L'évaluation consiste à mesurer la différence entre un résultat attendu et un résultat obtenu. Elle peut se faire en utilisant plusieurs mesures dont quelques unes sont :

- **Matrice de contingence ou de confusion** : Cette technique utilise un corpus étiqueté de documents pour lequel on connaît la vraie catégorie de chaque document, et le résultat obtenu par le classifieur.Pour un corpus, on construit la matrice de contingence pour chaque classe , qui fournit 4 informations essentielles :
  - **Vrai Positif (VP)** : documents correctement classés ;
  - **Vrai Négatif(VN)** : documents correctement non classés ;
  - **Faux Positif (FP)** : documents incorrectement classés ;
  - **Faux Négatif (FN)** : documents incorrectement non classés.

Classification des documents

Évaluation d'un classificateur

- **Matrice de confusion d'une classe**

Catégorie C_i		**Jugement expert**	
		Oui	Non
Jugement classifieur	Oui	**VPi**	**FPi**
	Non	**FNi**	**VNi**

- **Matrice de confusion de plusieurs classes**

		Expert	
	$C = \{c_1, c_2, ..., c_{\|C\|}\}$	C_i	$\neg C_i$
Classifieur	C_i	$VP = \sum_{i=1}^{\|C\|} VP_i$	$FP = \sum_{i=1}^{\|C\|} FP_i$
	$\neg C_i$	$FN = \sum_{i=1}^{\|C\|} FN_i$	$VN = \sum_{i=1}^{\|C\|} VN_i$

- **Le Rappel** : Elle est la proportion de documents correctement classés par le système par rapport à tous les documents de la classe Ci, elle mesure la capacité d'un système de classification à détecter les documents correctement classés . Elle est donnée par la relation :

$$\begin{aligned}\textbf{Rappel}(C_i) &= \frac{\textbf{Nombrededocumentsbienclassésdans}\textbf{C}_i}{\textbf{Nombrededocumentsdela}\textbf{C}_i} \\ &= \frac{\textbf{VP}_i}{\textbf{VP}_i + \textbf{FN}_i}\end{aligned}$$

- **La précision :** est la proportion de documents correctement classés parmi ceux classés par le système dans Ci .Elle mesure la capacité d'un système de classification à ne pas classer un document dans une classe, un document qui ne l'est pas. Comme elle peut aussi être interprétée par la probabilité conditionnelle qu'un document choisi aléatoirement dans la classe soit bien classé par le classifieur. Elle est donnée par la relation

$$\begin{aligned}\textbf{Précision}(C_i) &= \frac{\textbf{Nombredededocumentsbienclassésdans}\textbf{C}_i}{\textbf{Nombredededocumentsclassésdans}\textbf{C}_i} \\ &= \frac{\textbf{VP}_i}{\textbf{VP}_i + \textbf{FP}_i}\end{aligned}$$

Calcul de précision et rappel pour plusieurs classes

$$\boldsymbol{P} = \frac{VP}{VP+FP} = \frac{\sum_{i=1}^{|C|} VP_i}{\sum_{i=1}^{|C|} \left(VP_i + FP_i\right)}$$

$$\boldsymbol{R} = \frac{VP}{VP+FN} = \frac{\sum_{i=1}^{|C|} VP_i}{\sum_{i=1}^{|C|} \left(VP_i + FN_i\right)}$$

- **Le bruit** est le pourcentage de textes incorrectement associés à une classe par le système. Il est donné par la formule :

$$\textbf{Bruit(B)} = \textbf{1-Précision(P)} = \frac{\textbf{FP}_i}{\textbf{VP}_i + \textbf{FP}_i}$$

- **Le silence** est le pourcentage de textes à associer à une classe incorrectement non classés par le système. Il est donné par la formule

$$\textbf{Silence(S)} = \textbf{1-Rappel(R)} = \frac{\textbf{FN}_i}{\textbf{VP}_i + \textbf{FN}_i}$$

Classification des documents

Évaluation d'un classificateur

- La **F-mesure** est le plus usuel des indicateurs. Elle prend en compte la valeur relative de la précision et du rappel.Elle est calculée par la formule :

$$\textbf{F-measure} = \frac{2 \times \textbf{précision} \times \textbf{rappel}}{\textbf{précision} + \textbf{rappel}}$$

- **Le taux de succès** est le rapport entre les documents bien classés sur le nombre total des documents du corpus.Il se calcule par la formule :

$$\textbf{Taux succès} = \frac{\textbf{VP} + \textbf{VN}}{\textbf{VP} + \textbf{FP} + \textbf{VN} + \textbf{FN}}$$

- **Taux d'erreur :** est le rapport entre les documents mal classés sur le nombre total des documents du corpus. Il se calcule par la formule :

$$\textbf{Taux d'erreur} = 1 - \textbf{Taux succès} = \frac{\textbf{FP} + \textbf{FN}}{\textbf{VP} + \textbf{FP} + \textbf{VN} + \textbf{FN}}$$

Classification des documents

Problèmes

La classification automatique de texte, tout comme d'autres domaine de la science est soumis lui aussi à plusieurs contraintes qui sont, d'ailleurs, essentielles à l'essor de cette discipline. Quelques uns sont :

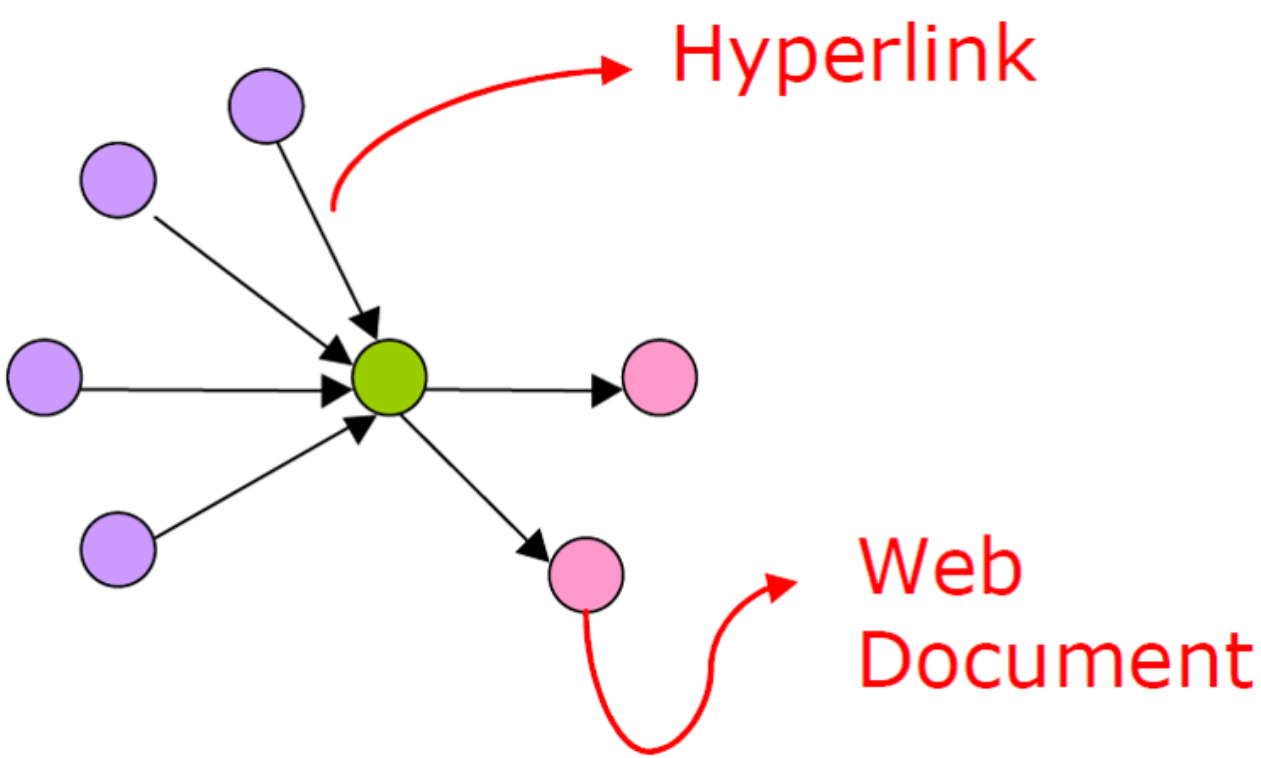
1. La polysémie
2. La redondance sémantique
3. Le temps d'apprentissage
4. Problème d'étiquetage

CHAPITRE 4

Le Web Structure Mining

Introduction

La structure d'un graphique Web typique se compose de pages Web en tant que nœuds et d'hyperliens en tant que bords reliant deux pages liées.



Introduction

Web Structure Mining

Encore appelé l'exploration de structure du Web peut être défini comme le processus de découverte d'informations à partir de la structure du Web. Il traite principalement de :

1. découverte du modèle sous-jacent à la structure des liens du Web
2. la topologie des hyperliens avec ou sans la description des liens

- Ce type d'exploration peut être effectué soit au niveau du document (intra-page) soit au niveau du lien hypertexte (inter-page)
- La recherche au niveau des hyperliens est également appelée analyse des hyperliens

Introduction

L'importance d'etude de la structure des hyperliens :

1. Les hyperliens ont deux objectifs principaux.
   - Navigation pure.
   - Pointez vers des pages faisant autorité sur le même sujet que la page contenant le lien.
2. Cela peut être utilisé pour récupérer des informations utiles sur le Web.

Pourquoi le Web Structure Mining ?

- les modèles peuvent être utilisés pour classer les pages Web
- utiles pour créer des informations telles que la similitude et la relation entre les sites Web
- utile pour découvrir le type de site.

Terminologie du Web structure Mining

Définitions des termes

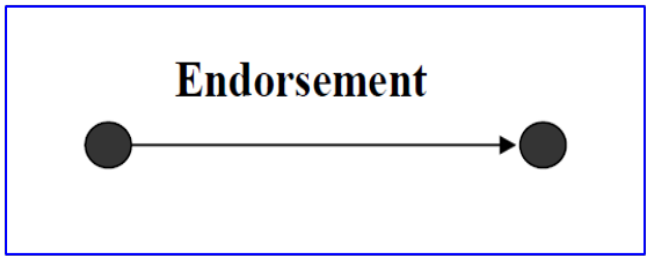
- **Web-Graph** : un graphique orienté qui représente le Web.
- **Nœud (Node)** : Chaque page Web est un nœud du Web-graph.
- **Lien** : Chaque lien hypertexte sur le Web est une arête dirigée du Web-graph.
- **Degré interieur (In-degree)** : Le degré interieur d'un nœud, p est le nombre de liens distincts qui pointent vers p.
- **Degré extérieur (Out-degree)** : Le degré extérieur d'un nœud, p est le nombre de liens distincts provenant de p qui pointent vers d'autres nœuds.
- **Chemin dirigé (Directed Path)** : une séquence de liens, à partir de p, qui peut être suivie pour atteindre q.

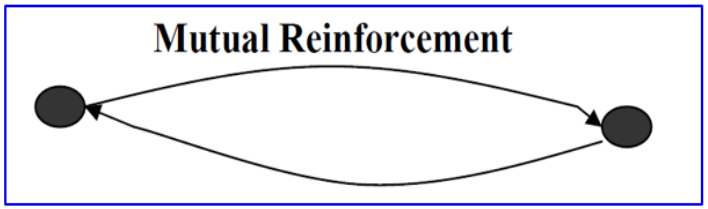
Terminologie du Web structure Mining

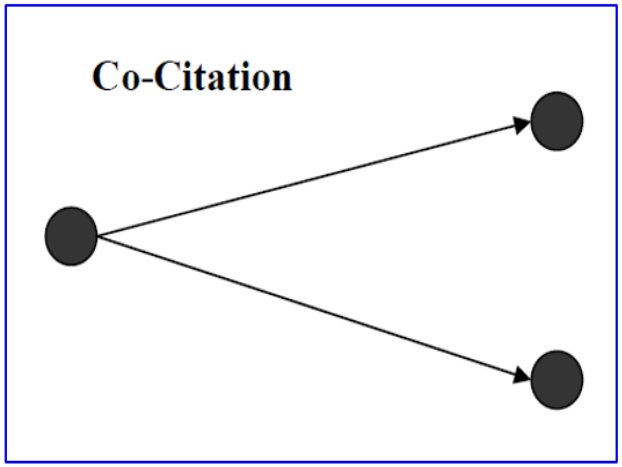
Définitions des termes

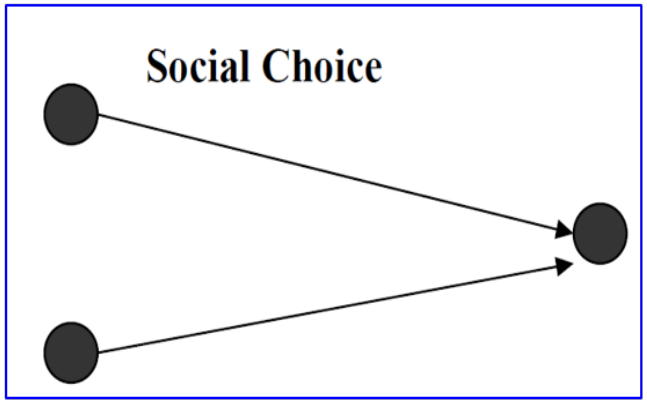
- **Chemin le plus court (Shortest Path)** : de tous les chemins entre les nœuds p et q, lequel a la longueur la plus courte, c'est-à-dire le nombre de liens qu'il contient.
- **Diamètre(Diameter)** : Le maximum de tous les chemins les plus courts entre une paire de nœuds p et q, pour toutes les paires de nœuds p et q dans le graphe Web.

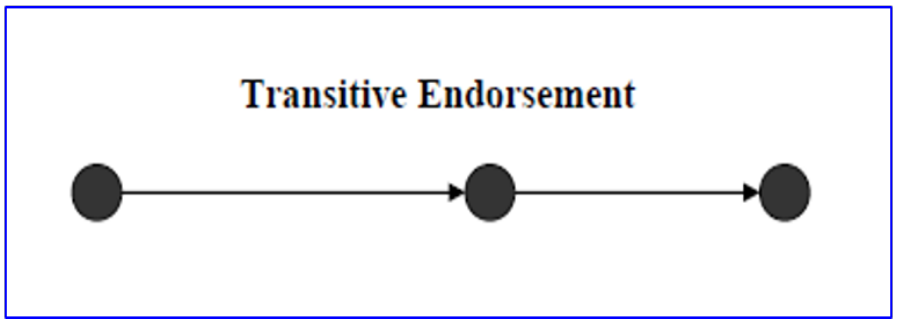
Quelques Structures Web



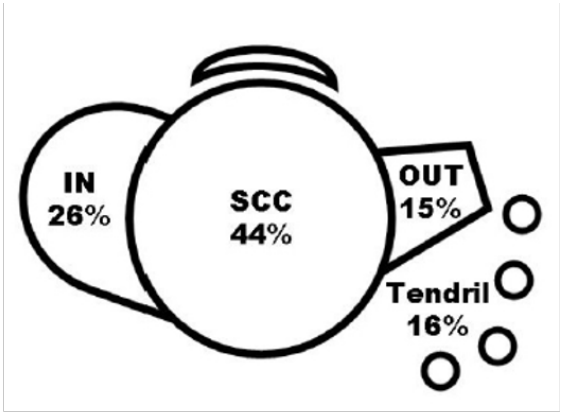




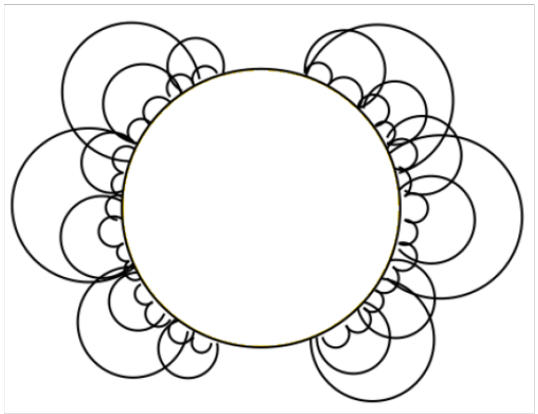




Les formes du Web



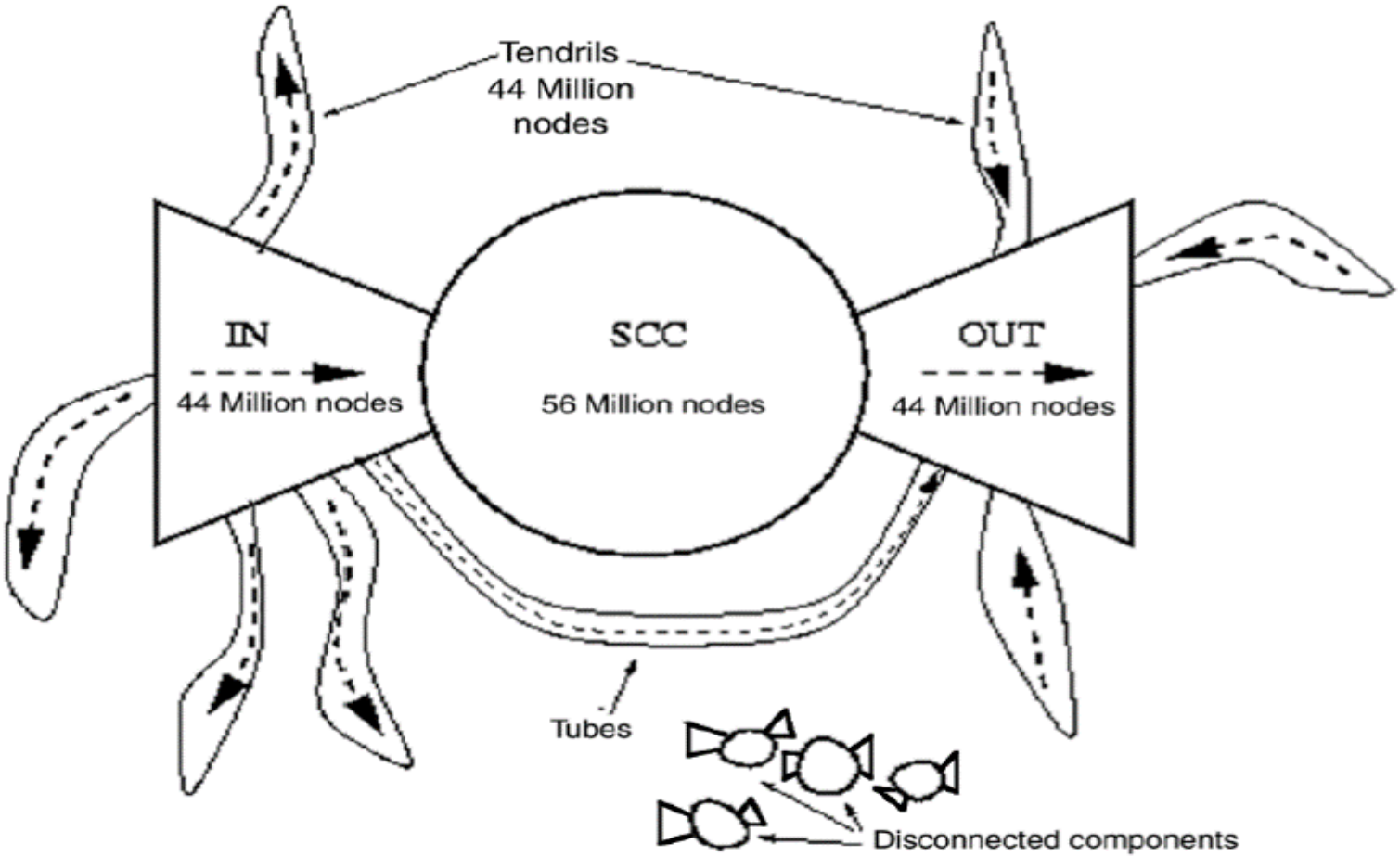
La forme du graphique Web Chinois



La forme graphique du Web représentée plus précisément par un graphique en forme de marguerite

Le modèle en nœud papillon du Web

- Le nom a en fait été donné par Andrei Broder lorsque ses collègues essayaient de donner un sens aux données Web collectées.



Le modèle en nœud papillon du Web

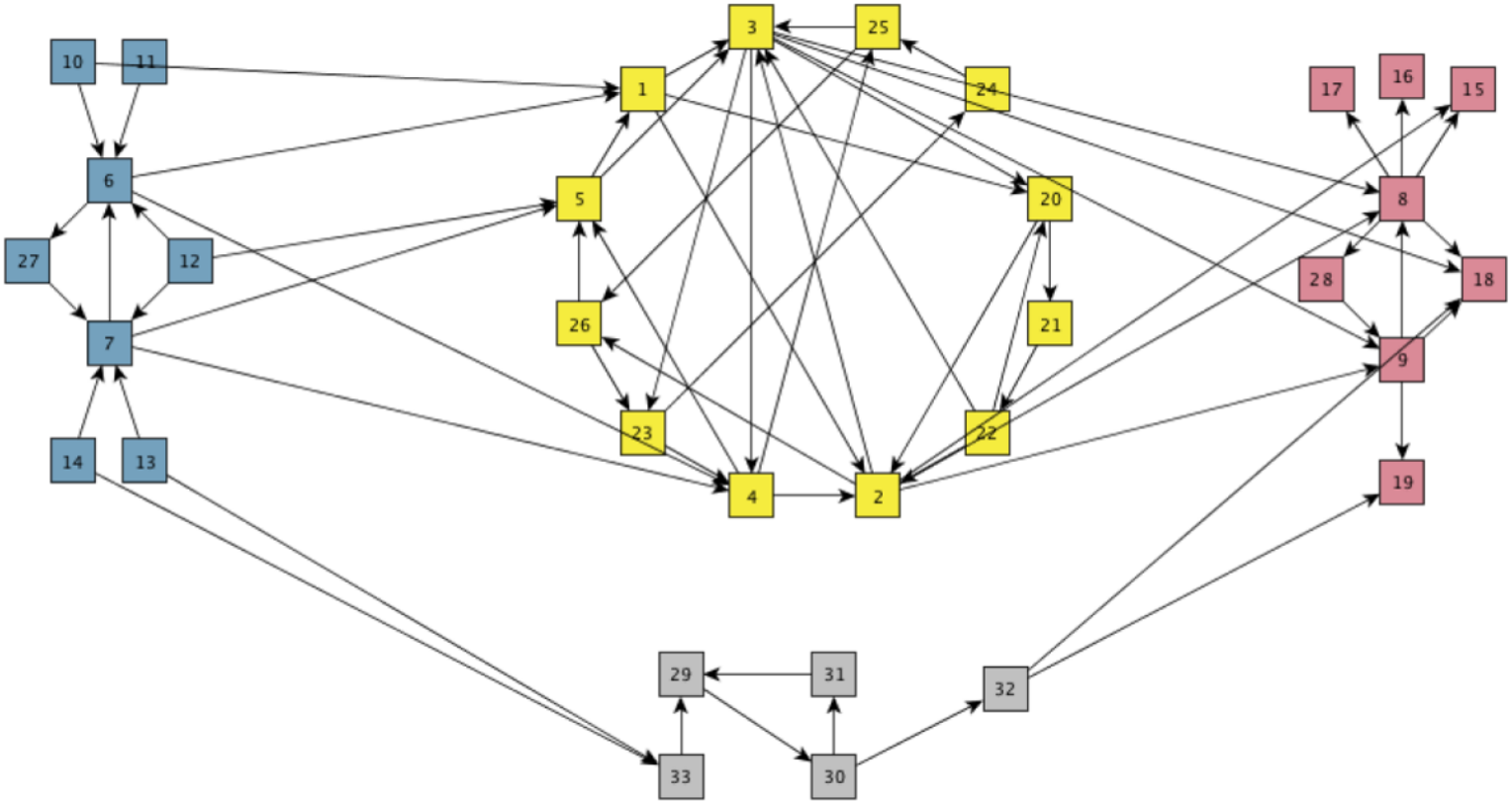
- **SCC (noyau)** : noyau fortement connecté
- **IN** est composé des nœuds qui se trouvent sur un chemin dirigé qui se termine sur un nœud dans CORE, mais qui eux-mêmes ne font pas partie du CORE.
- **OUT** est composé des nœuds qui se trouvent sur un chemin dirigé qui part d'un nœud dans CORE, mais qui eux-mêmes ne font pas partie du CORE.
- **ISLANDS** (Les ÎLES) sont des nœuds complètement déconnectés de CORE, IN et OUT, c'est-à-dire qu'il n'y a pas de chemin dirigé qui les relie au Bowtie(Modèle en nœud papillon).

Le modèle en nœud papillon du Web

Les **TENDRILS** se déclinent en trois saveurs :

- **TENDRILS-IN** sont des nœuds pour lesquels il existe un chemin dirigé depuis **IN**, mais il n'y a pas de chemin dirigé d'eux vers un autre composant.
- **TENDRILS-OUT** sont des nœuds qui sont sur un chemin dirigé vers un nœud dans **OUT**, mais aucun chemin ne mène d'eux à un autre composant.
- Les **TUBES** sont des nœuds qui se trouvent sur un chemin allant d'un nœud dans **IN** à un nœud dans **OUT**, et il n'y a pas de chemin qui les relie à **CORE**.

Le modèle en nœud papillon du Web

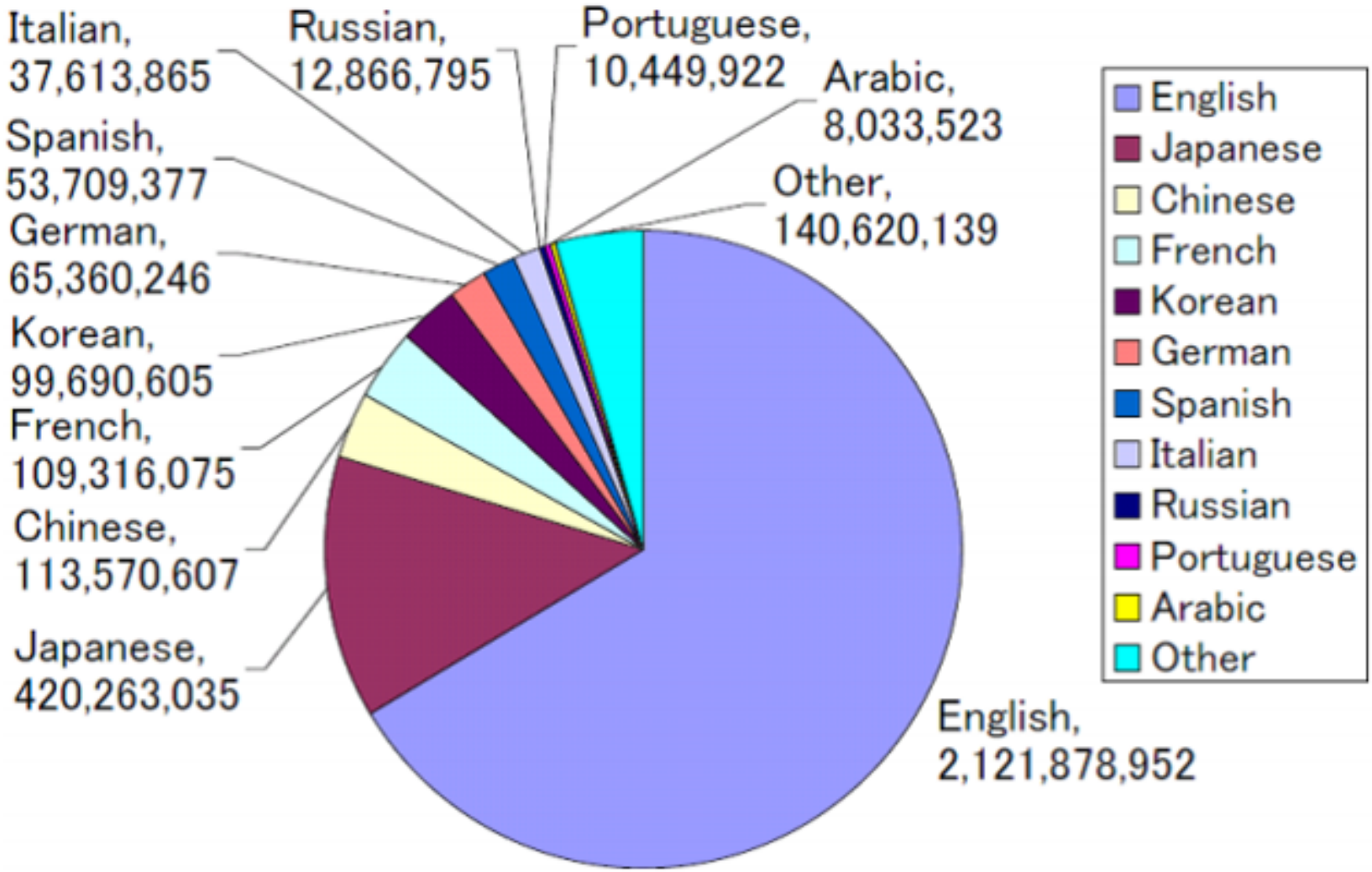


Le modèle en nœud papillon du Web

S'il n'y a pas de **CORE**, il n'y a pas de graphique Bow-Tie (Modèle en nœud papillon). Mais il est pratiquement impossible que la collection de pages Web interconnectées dans le Web Graph n'ait pas de SCC. En effet, étant donné un ensemble de pages Web autorisées à lien une ou plusieurs fois vers n'importe quelle autre page Web de la collection qu'ils choisissent, un SCC est inévitable, et avec lui un nœud papillon.

Le modèle en nœud papillon du Web

Example : web structure by language



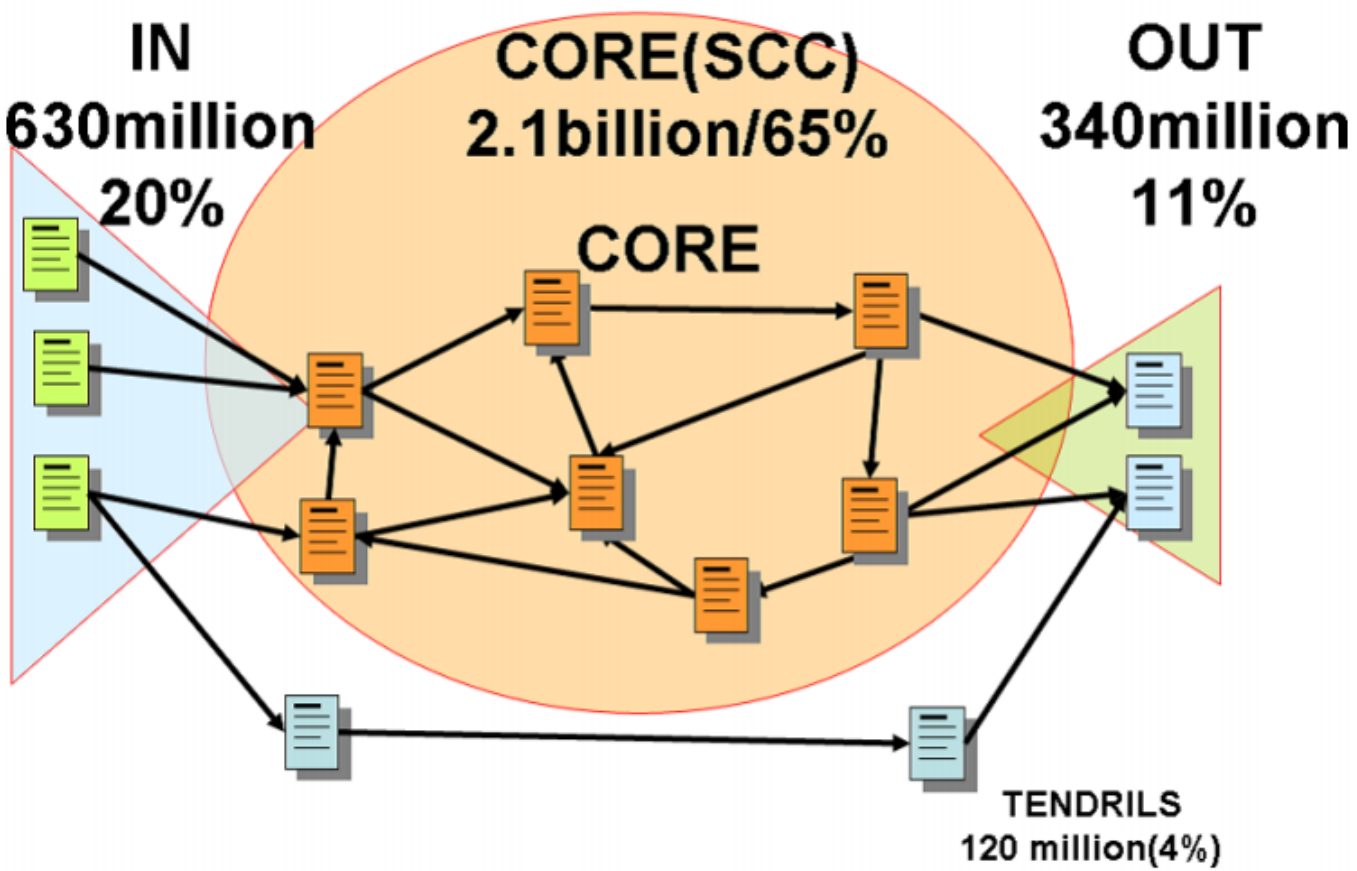
Le modèle en nœud papillon du Web

Example : Components of web structures by Language

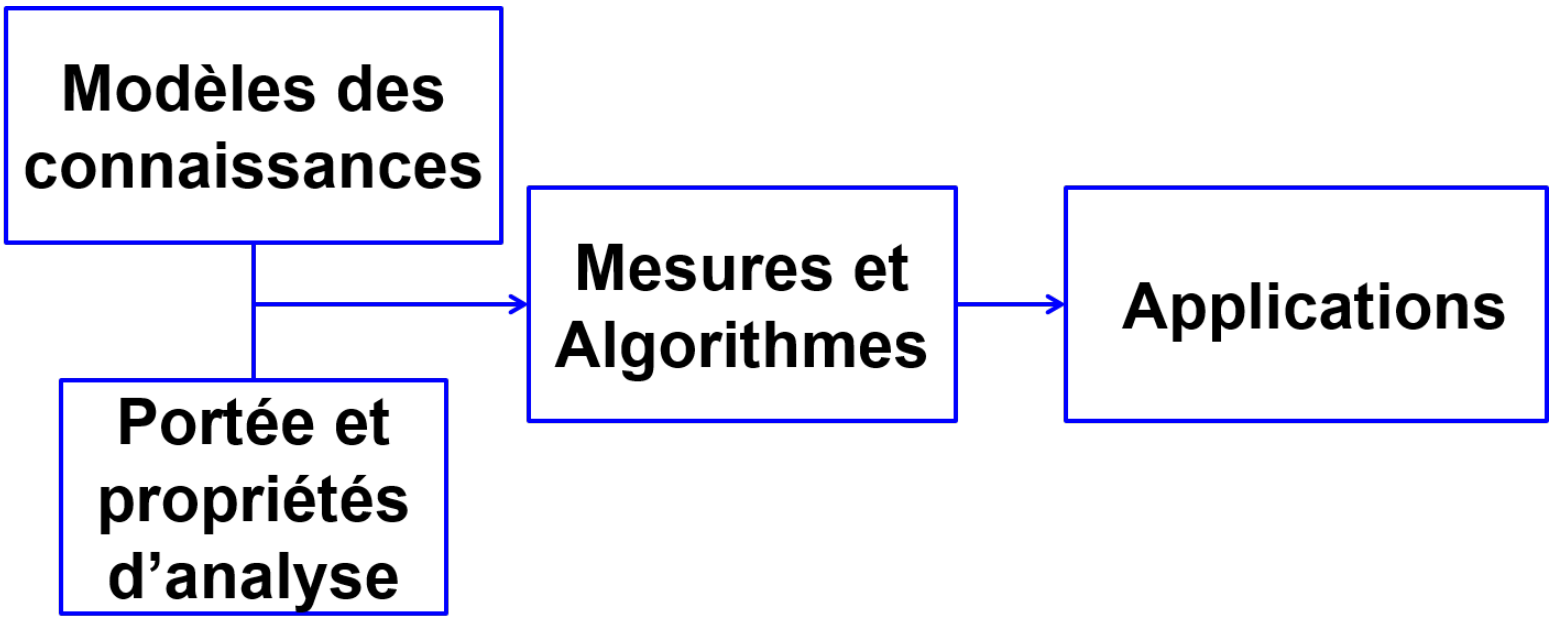
TLD	CORE	IN	OUT	Other
Chinese	76.88%	9.98%	10.57%	2.57%
Japanese	71.05%	25.85%	2.54%	0.56%
English	66.90%	9.04%	16.44%	7.62%
Spanish	64.93%	5.30%	23.60%	6.16%
French	61.85%	9.23%	20.65%	8.27%
Arabic	61.43%	10.20%	18.59%	9.78%
Korea	54.32%	17.07%	19.36%	9.25%
Russian	35.76%	18.20%	18.35%	27.69%
Portuguese	26.60%	4.94%	42.18%	26.28%
German	26.61%	8.16%	42.18%	23.05%
Italian	23.67%	17.10%	29.54%	29.69%
Other	7.24%	1.98%	9.32%	81.47%


Le modèle en nœud papillon du Web

Example :



Techniques d'analyses d'hyperliens



Techniques d'analyses d'hyperliens

- **Modèles de connaissances** : les représentations sous-jacentes qui constituent la base de l'exécution de la tâche spécifique à l'application
- **Portée et propriétés de l'analyse** : la portée de l'analyse spécifie si la tâche est pertinente pour un nœud unique ou un ensemble de nœuds ou pour l'ensemble du graphique. Les propriétés sont les caractéristiques d'un seul nœud ou de l'ensemble de nœuds ou du Web entier.
- **Mesures et algorithmes** : les mesures sont les normes des propriétés telles que la qualité, la pertinence ou la distance entre les nœuds. Les algorithmes sont conçus pour un calcul efficace de ces mesures.

Ces trois domaines forment les blocs fondamentaux pour la construction de diverses applications basées sur l'analyse des liens hypertexte.

Algorithmes de Web Structure Mining

1. **Algorithme PageRank (Google Founders)**
   - Examine le nombre de liens vers un site Web et l'importance des liens de référence
   - calculé avant que l'utilisateur ne saisisse la requête.
2. **Algorithme HITS (Hyperlinked Induced Topic Search)**
   - L'utilisateur reçoit deux listes de pages à interroger (pages d'autorité et de liens)
   - Les calculs sont effectués une fois que l'utilisateur a saisi la requête.

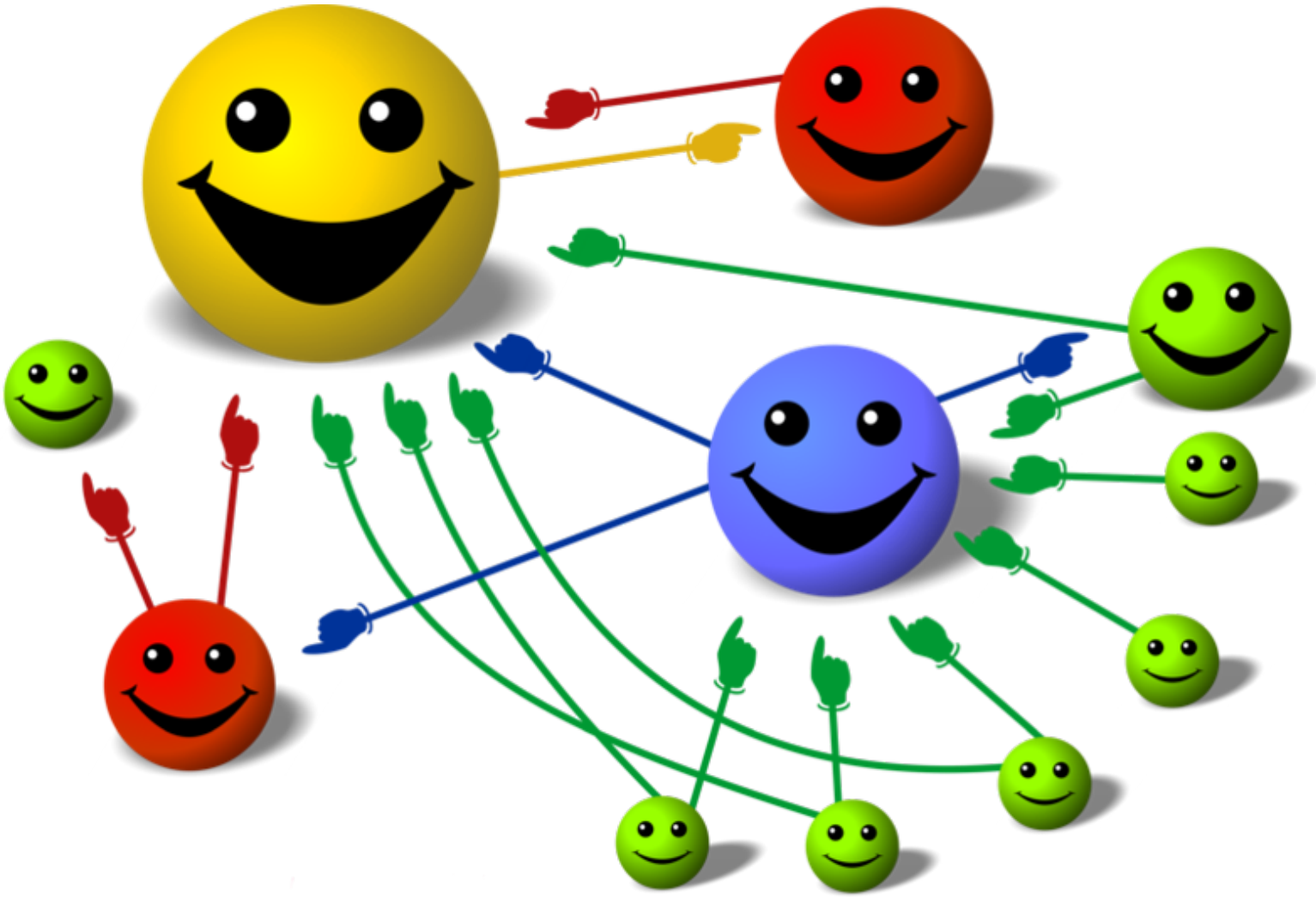
Algorithme Google's PageRank

- L'idée de l'algorithme est venue de la littérature de citation académique.
- Il a été développé en 1998 dans le cadre du prototype du moteur de recherche Google
- Étudie la relation de citation des documents sur le Web.
- Le moteur de recherche Google classe les documents en fonction à la fois des termes de la requête et de la structure des hyperliens du Web.

Définition du PageRank

- Le **PageRank** produit un classement indépendant de la requête d'un utilisateur.
- L'importance d'une page Web est déterminée par le nombre d'autres pages Web importantes qui pointent vers cette page et le nombre de liens sortants d'autres pages Web.

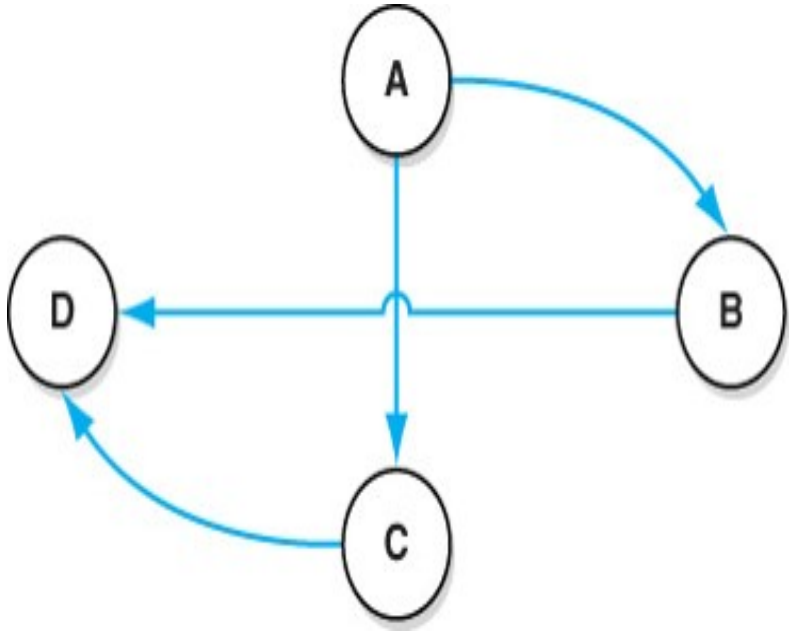
Algorithme Google's PageRank



Algorithme Google's PageRank

Exemple des liens de retour :

La page A est un backlink de la page B et de la page C, tandis que la page B et la page C sont des backlinks de la page D.

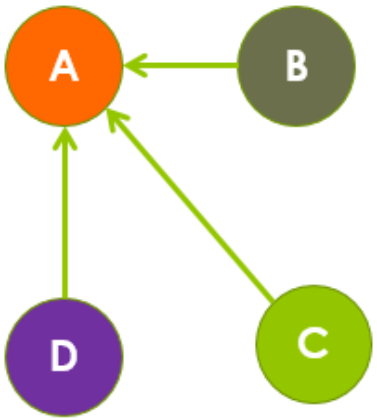


Backlink = Outlink= OutDegree

Algorithme Google's PageRank

Exemple 1 :

$$PR(A) = \frac{PR(B)}{1} + \frac{PR(C)}{1} + \frac{PR(D)}{1}$$

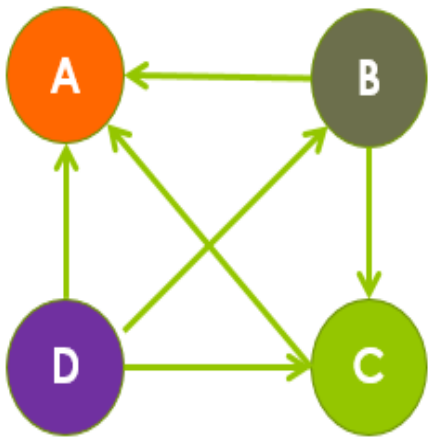


$$PR(A) = 0.25 + 0.25 + 0.25$$
$$PR(A) = 0.75$$

Algorithme Google's PageRank

Exemple 2 :

$$PR(A) = \frac{PR(B)}{2} + \frac{PR(C)}{1} + \frac{PR(D)}{3}$$



$$PR(A) = PR(B)/2 + PR(C)/1 + PR(D)/3$$
$$= 0.125 + 0.25 + 0.0833$$
$$= 0.4583$$

Algorithme Google's PageRank

Classement des pages :

Une page aura un rang de page élevé si :

- De nombreuses pages y font référence.
- Certaines pages y pointant ont des rangs de page élevés.

Autrement dit :

- Les pages bien situées sur le Web valent la peine d'être consultées.
- Les pages qui n'ont qu'une seule citation d'une page Web de haut niveau valent la peine d'être examinées.

Algorithme Google's PageRank

Facteur d'amortissement

- La théorie du PageRank soutient que même un internaute imaginaire qui clique au hasard sur des liens finira par arrêter de cliquer. La probabilité, à n'importe quelle étape, que la personne continue est un **facteur d'amortissement d**.
- Le **facteur d'amortissement** est soustrait de 1 et ce terme est ensuite ajouté au produit du facteur d'amortissement et de la somme des scores PageRank entrants.
- Ainsi, le PageRank de n'importe quelle page est dérivé en grande partie des PageRanks des autres pages. Le facteur d'amortissement ajuste la valeur dérivée vers le bas.

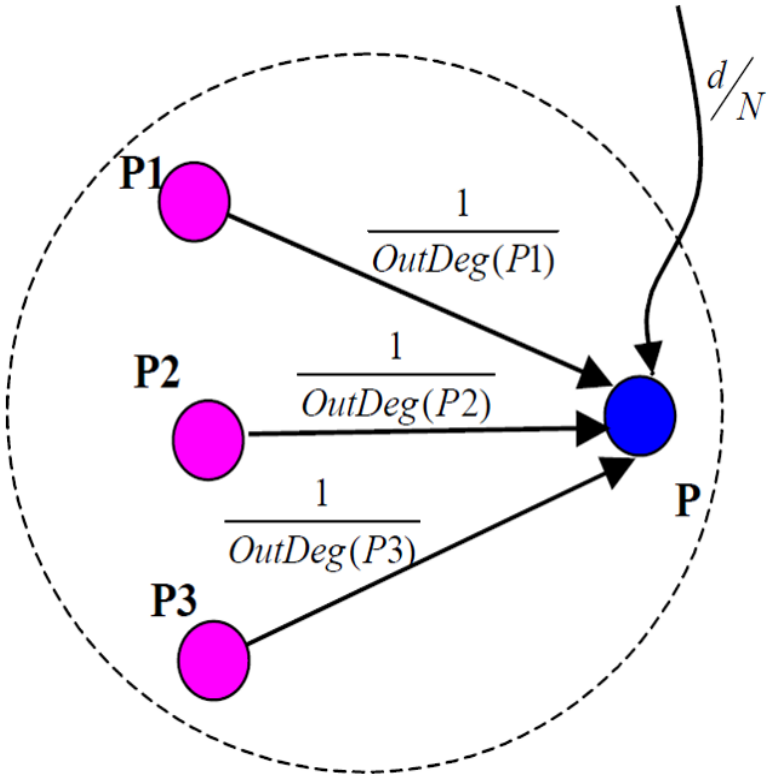

Algorithme Google's PageRank

Le PageRank d'une page u est calculé comme suit :

$$PageRank(u) = (1-d) + d \sum_{(v,u)\in E} \frac{PageRank(v)}{OutDegree(v)}$$

où, OutDegree(v) représente le nombre de liens sortant de la page v et le paramètre d un facteur d'amortissement, qui peut être un nombre réel compris entre 0 et 1. La valeur de d est généralement prise égale à 0,85.

Algorithme Google's PageRank



Idée clé : Le classement d'une page Web dépend du classement des pages Web qui y pointent

$$PR(P) = d/N + (1-d)\left(\frac{PR(P1)}{OutDeg(P1)} + \frac{PR(P2)}{OutDeg(P2)} + \frac{PR(P3)}{OutDeg(P3)}\right)$$

Algorithme Google's PageRank

Algorithme Google's PageRank

1. Supposons qu'il y ait n pages liées.
2. Soit $S = (V, E)$ où V est l'ensembles des pages et E l'ensemble des liens entre les pages.
3. Initialiser PageRank (p) $= 0$ pour toutes les pages.
4. Répéter jusqu'à ce que le vecteur PageRank converge (C'est-à-dire se stabiliser ou ne pas changer) :
   - Pour toutes pages $u \in V$
   - $$PageRank(u) = (1-d) + d \sum_{(v,u)\in E} \frac{PageRank(v)}{OutDegree(v)}$$
5. Retourner le vecteur PageRank

Algorithme Google's PageRank

Un réseau simple de pages (Ian Roger, 2006)

OutDegree(A) = 1 et OutDegree(B) = 1.
Ici, nous ne savons pas ce que devraient être leurs PageRanks pour commencer, nous pouvons donc faire une estimation à 1,0 , en supposant d=0,85, et effectuer les calculs suivants :

$$\textbf{PageRank(A)= (1 – d) + d (PageRank(B)/1)}$$
$$\textbf{PageRank(B)= (1 – d) + d (PageRank(A)/1)}$$

$$\textbf{PageRank(A)= 0,15 + 0,85 * 1 = 1}$$
$$\textbf{PageRank(B)= 0,15 + 0,85 * 1 = 1}$$

Nous avons calculé que le PageRank de A et B est de 1.

Algorithme Google's PageRank

Maintenant, nous insérons 0 comme estimation et effectuons à nouveau les calculs :

PageRank(A) = 0,15 + 0,85 * 0 = 0,15
PageRank(B) = 0,15 + 0,85 * 0,15= 0,2775

Nous avons maintenant une autre supposition pour le PageRank(A) donc nous l'utilisons pour calculer le PageRank(B) et continuons :

PageRank(A) = 0,15 + 0,85 * 0,2775 = 0,3859
PageRank(B) = 0,15 + 0,85 * 0,3859 = 0,4780

En répétant les calculs, on obtient :

PageRank(A) = 0,15 + 0,85 * 0,4780 = 0,5563
PageRank(B) = 0,15 + 0,85 * 0,5563 = 0,6229

Algorithme Google's PageRank

Si nous répétons les calculs, les PageRanks des deux pages convergent finalement vers 1.

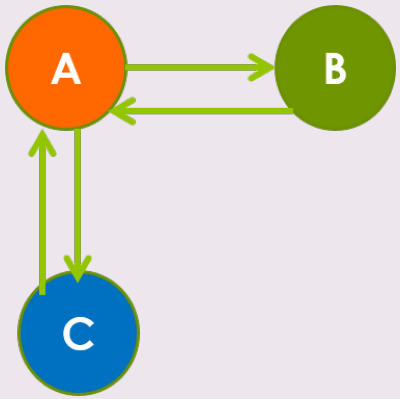
Remarques sur l'algorithme PageRank :

Une page sans successeurs n'a aucune portée pour envoyer son importance. De même, un groupe de pages qui n'ont aucun lien hors du groupe finira par recueillir toute l'importance du Web.

Algorithme Google's PageRank

Exercice :

Soit le Web graph ci-dessous :



Il existe un lien entre la page A vers B et C. Il existe également un lien entre les pages B et C vers A.

Calculer PageRank(A), PageRank(B) et PageRank(C)

- Commencez avec la valeur initiale de PageRank à 0.
- Compléter 6 itérations

CHAPITRE 5

Le Web Usage Mining

Introduction

- La croissance et la prolifération continues du commerce électronique
- des services Web et des systèmes d'information basés sur le Web
- les volumes de flux de clics
- de données de transaction et de données de profil d'utilisateur collectées par les organisations basées sur le Web dans leurs opérations quotidiennes

ont atteint des proportions astronomiques.

Introduction

L'analyse de ces données peut aider ces organisations à :

- déterminer la valeur à vie des clients,
- concevoir des stratégies de marketing croisé pour les produits et services,
- évaluer l'efficacité des campagnes promotionnelles,
- optimiser la fonctionnalité des applications Web,
- fournir un contenu plus personnalisé aux visiteurs et trouver la structure logique la plus efficace pour leur espace Web

Ce type d'analyse implique la **découverte** automatique de **modèles** et de **relations significatifs** à partir d'une vaste collection de données principalement semi-structurées, souvent stockées dans des journaux d'accès aux serveurs Web et d'applications, ainsi que dans des sources de données opérationnelles connexes.

Introduction

Le **WUM** fait référence à la découverte et à l'analyse automatiques des modèles de flux de clics, des transactions des utilisateurs et d'autres données associées collectées ou générées à la suite des interactions des utilisateurs avec les ressources Web sur un ou plusieurs sites Web.
L'objectif est de **capturer**, **modéliser** et **analyser** les schémas comportementaux et les profils des utilisateurs interagissant avec un Site Internet.
Les modèles découverts sont généralement représentés sous forme de collections de pages, d'objets ou de ressources fréquemment consultés ou utilisés par des groupes d'utilisateurs ayant des besoins ou des intérêts communs.

Définitions

Définitions

- **Ressource** - d'après la spéciation de W3C pour URI, une ressource R peut être tout objet ayant une identité Comme exemples de ressources, nous pouvons citer : un chier html, une image ou un service Web.
- **Ressource Web** - une ressource accessible par une version du protocole HTTP ou un protocole similaire (ex. HTTP-NG).
- **Serveur Web** - un serveur qui donne accès à des ressources Web.
- **Requête Web** - une requête pour une ressource Web, faite par un client (navigateur Web) à un serveur Web
- **Page Web** - ensemble des informations, consistant en une (ou plusieurs) ressource(s) Web, identifiée(s) par un seul URI.

Définitions

Définitions

- **Vue de page** (page view) - le fait d'afficher une page Web dans l'environnement visuel client à un moment précis en temps. Une vue de page (ou page) peut être composée de plusieurs pages Web et ressources Web comme, par exemple, dans le cas des pages incluant des cadres
- **Navigateur Web** (Browser) - logiciel de type client chargé d'afficher des pages à l'utilisateur et de faire des requêtes HTTP au serveur Web.
- **Utilisateur** - personne qui utilise un navigateur Web.
- **Session utilisateur** - un ensemble délimité des clics utilisateurs sur un (ou plusieurs) serveur(s) Web.
- **Visite(s)** - L'ensemble des clics utilisateur sur un seul serveur Web (ou sur plusieurs) pendant une session utilisateur.

Définitions

Définitions

Episode - un sous-ensemble de clics liés qui sont présents dans une session utilisateur. C'est encore une sous-séquence d'une session composé des pages vues sémantiquement ou fonctionnellement liées.

Exemple : pendant une session sur **profsinfocmr.org**, l'utilisateur a vérifié son e-mail, a regardé les photos des membres du groupe PIC et a téléchargé les épreuves. Il s'agit dans ce cas de trois épisodes distincts.

Les fichiers Logs

Chaque demande d'affichage d'une page Web, de la part d'un utilisateur, peut générer plusieurs requêtes.Des informations sur ces requêtes (notamment les noms des ressources demandées et les réponses du serveur Web) sont stockées dans les fichiers appelés "**fichiers logs**"

Ces fichiers logs peuvent être stockés dans 3 endroits différents :

1. **Serveur web** : fichiers journaux
2. **Serveur proxy** : requêtes http du client
3. **Navigateur du client** : les cookies

Les fichiers Logs

Il existe plusieurs types des fichiers logs

- **Fichiers logs d'accès** : enregistrent toutes les demandes traitées par le serveur
- **Fichiers logs d'erreurs** : enregistrent les incidents survenus lors du dialogue avec le serveur
- **Fichiers logs référentiels** : indiquent les sites, les pages de provenance et d'arrivées
- **Fichiers logs agents** : enregistrent les informations sur l'utilisateur (caractéristiques du navigateur, système d'exploitation...)

Les fichiers Logs

Il existe plusieurs formats pour les fichiers logs Web, mais le plus courant est le CLF (Common LogFile Format). Selon ce format six informations sont stockées :

1. **Le nom** ou **l'adresse IP** de la machine appelante
2. le **nom** et le **login** HTTP de l'utilisateur
3. la **date** et **l'heur**e de la requête
4. la **méthode** utilisée dans la requête (GET, POST, etc.) et le n**om de la ressource Web** demandée
5. le **statut** de la requête ou code d'erreur qui indique si l'action s'est bien déroulée (prend la valeur 200 en cas de réussite de la requête)
6. la **taille** (size) du fichier envoyé.

Les fichiers Logs

Le format ECLF (Extended Common Log Format), qui représente une version plus complète du CLF, contient en plus :

- le nom du **navigateur Web**
- le **système d'exploitation** (cf.User ou Agent)
- l'**adresse de la page** où se trouvait avant l'utilisateur, lorsqu'il a lancé la requête (referrer).

Une ligne d'un log ECLF est présenté ci-dessous :

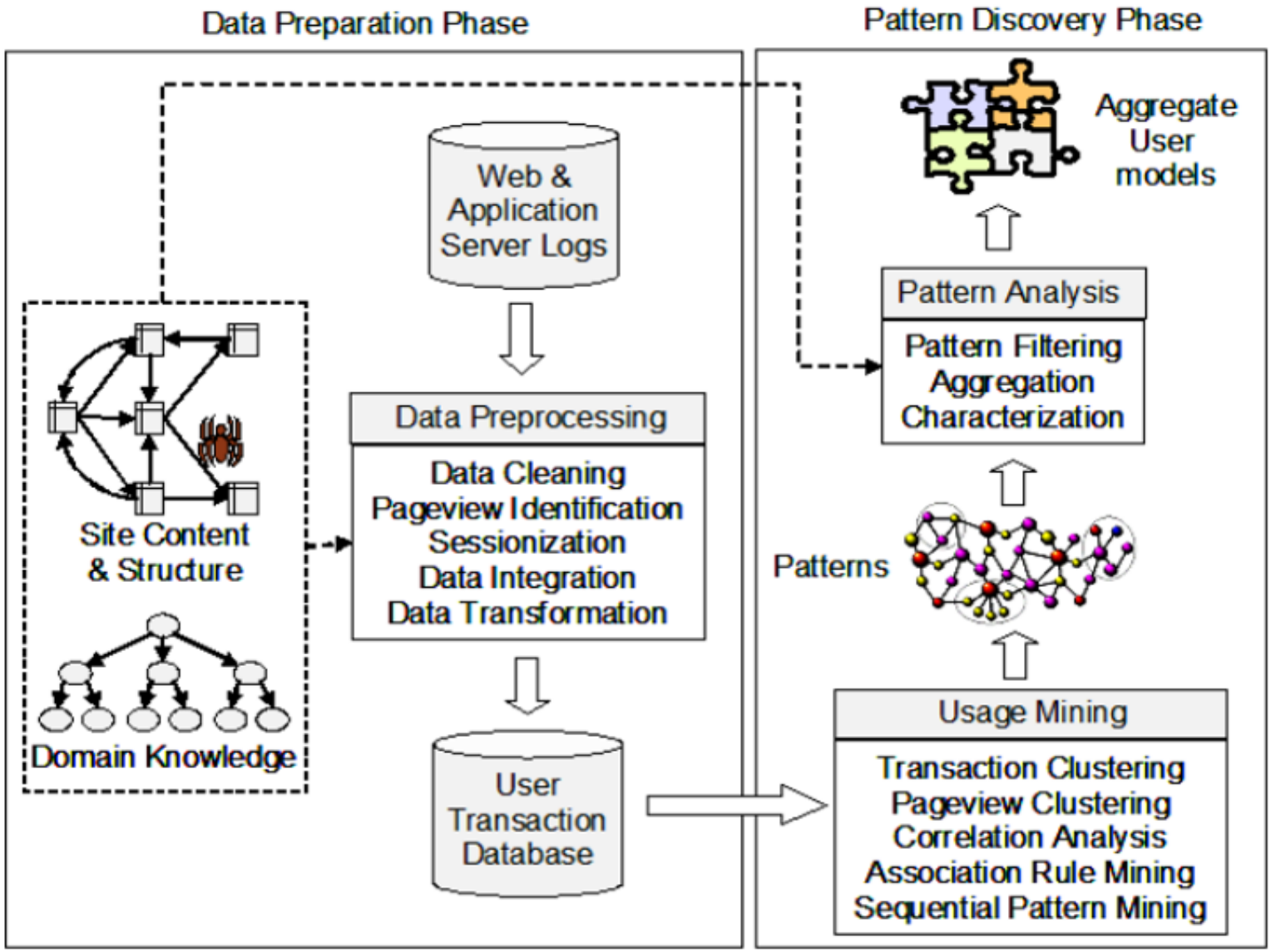
```
138.96.69.7 - - [30/May/2003 :16 :43 :58 +0200] "GET
/axis/people.shtml HTTP/1.0" 200 10677 "http ://www-
sop.inria.fr/axis/table.html" "Mozilla/4.76 [en] (X11 ; U ; Linux
2.4.20 i686)"
```


Les fichiers Logs

Exercice :

1. Présenter d'autres format de fichiers logs en dehors de ceux présentés dans ce cours.
2. Présenter d'autres valeurs de types de requêtes dans un fichier log en dehors de POST et GET.
3. Donner d'autres valeurs du champ « statut » dans un fichier log puis donner leur signification.
4. Décomposer la ligne du fichier log donné ci-dessous :
   129.0.210.220 - - [24/Jul/2022 :09 :05 :00 +0200] "GET /img/portfolio/6.jpg HTTP/2.0" 301 333 "https ://profsinfocmr.org/" "Mozilla/5.0 (Linux ; Android 8.1.0 ; TECNO CF7k) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/103.0.0.0 Mobile Safari/537.36"

Processus du Web Usage Mining



Processus du Web Usage Mining

☐ Collecte des données :

La première étape est la collecte des données à exploiter qui consiste à rassembler les données du web qui vont être analysées.
Les principales sources de données utilisées dans WUM sont

- les **fichiers journaux du serveur**, qui incluent les journaux d'accès au serveur Web et les journaux du serveur d'applications.
- Les sources de **données supplémentaires** : les fichiers et les méta-données du site, les bases de données opérationnelles, modèles d'application et connaissance du domaine.

Processus du Web Usage Mining

☐ Prétraitement des données :

Le pré-traitement a comme objectif la structuration et l'amélioration de la qualité des données contenues dans les fichiers pour les préparer à une analyse des usagers.
Cette étape est souvent la plus coûteuse en termes de temps à cause de l'absence de structuration et la présence du bruit dans les données brutes. Les objets à identifier dans cette étape sont les requêtes effectuées par les humains, ainsi que par les robots web.
Le pré-traitement passe par deux phases principales :

1. **Une phase de nettoyage** : Elle consiste à supprimer les requêtes pour les ressources Web (images, fichiers multimédia, scripts pour le téléchargement) qui ne font pas l'objet de l'analyse et les requêtes ou visites provenant des robots Web, ne garder que les requêtes de type GET et les requêtes reflétant les pages web ayant une extension .html,.htm,.php..

Processus du Web Usage Mining

5. **une phase de transformation des données.**
   Elle regroupe les tâches suivantes :
   - Fusion des fichiers logs
   - Stockage des données dans une BD
   - Structuration des données
   - Identification des utilisateurs
   - Identification des vues de page
   - Identification des sessions
   - Identification des visites
   - Identification des épisodes

Processus du Web Usage Mining

Identification des utilisateurs

Pour regrouper les requêtes, il est important de savoir quels utilisateurs les ont émises. L'identification des utilisateurs à partir des fichiers logs tiennent compte de plusieurs facteurs :

- serveurs proxy,
- les adresses IP,
- le login,
- les cookies,
- les pages Web dynamiques,
- l'agent de l'utilisateur

Processus du Web Usage Mining

Identification des utilisateurs

Time	IP	URL	Ref	Agent
0:01	1.2.3.4	A	-	IE5;Win2k
0:09	1.2.3.4	B	A	IE5;Win2k
0:10	2.3.4.5	C	-	IE6;WinXP;SP1
0:12	2.3.4.5	B	C	IE6;WinXP;SP1
0:15	2.3.4.5	E	C	IE6;WinXP;SP1
0:19	1.2.3.4	C	A	IE5;Win2k
0:22	2.3.4.5	D	B	IE6;WinXP;SP1
0:22	1.2.3.4	A	-	IE6;WinXP;SP2
0:25	1.2.3.4	E	C	IE5;Win2k
0:25	1.2.3.4	C	A	IE6;WinXP;SP2
0:33	1.2.3.4	B	C	IE6;WinXP;SP2
0:58	1.2.3.4	D	B	IE6;WinXP;SP2
1:10	1.2.3.4	E	D	IE6;WinXP;SP2
1:15	1.2.3.4	A	-	IE5;Win2k
1:16	1.2.3.4	C	A	IE5;Win2k
1:17	1.2.3.4	F	C	IE6;WinXP;SP2
1:26	1.2.3.4	F	C	IE5;Win2k
1:30	1.2.3.4	B	A	IE5;Win2k
1:36	1.2.3.4	D	B	IE5;Win2k

User 1

0:01	1.2.3.4	A	-
0:09	1.2.3.4	B	A
0:19	1.2.3.4	C	A
0:25	1.2.3.4	E	C
1:15	1.2.3.4	A	-
1:26	1.2.3.4	F	C
1:30	1.2.3.4	B	A
1:36	1.2.3.4	D	B

User 2

0:10	2.3.4.5	C	-
0:12	2.3.4.5	B	C
0:15	2.3.4.5	E	C
0:22	2.3.4.5	D	B

User 3

0:22	1.2.3.4	A	-
0:25	1.2.3.4	C	A
0:33	1.2.3.4	B	C
0:58	1.2.3.4	D	B
1:10	1.2.3.4	E	D
1:17	1.2.3.4	F	C

Exemple d'identification des utilisateurs

Processus du Web Usage Mining

Identification des vues de page

Une vue de page p_i est composée de $p_i = \{r_{i1}, r_{i2}, ..., r_{iP}\}$ où $r_{ij} \in R(Ressources)$.

1. La requête pour la vue de page p_i est présente dans le fichier log. Dans ce cas, les entrées dans le fichier log qui correspond aux ressources contenues en p_i doivent être supprimées du fichier log et seulement la requête pour p_i est gardée.
2. La requête pour la page p_i manque (à cause du cache du navigateur Web ou d'un serveur de cache) et seulement quelques unes des entrées pour les ressources inclues en p_i sont présentes. Les entrées qui correspondent aux ressources doivent être remplacées par une requête pour p_i. Le temps de cette requêtes est mis égal à $t_i = min\{time(l_i)\}$, où l_i est l'entrée dans le fichier log qui correspond à la ressource r_i.

Formalisation d'une visite et session

- Étant donnée un intervalle de temps Δt, la visite v_{ij} de l'utilisateur u_i est définie : $v_{ij} =< u_i, t, p_{vi} >$, où $p_{vi} =< (t_1, p_1), (t_2, p_2)..., (t_n, p_n) >$, $t_{i+1} \geqslant t_i$ et $t_{i+1} - t_i < \Delta t, i = \overline{1..n-1}$
- La session s_i de l'utilisateur u_i est : $s_i = \{v_{ij}\}$, où v_{ij} est une visite de l'utilisateur u_i.

Pour un utilisateur u_i nous avons une séquence de vues de pages $< p_{ij} >$. Ceci représente sa séquence de clics sur un site Web (session serveur) dans une certaine période.

Processus du Web Usage Mining

Identification des visites et sessions

Par exemple, considérons l'utilisateur u qui a généré la session serveur suivante : $v = \{u, 16:09:10, <(A, 16:09:10), (B, 16:09:43), (C, 16:12:02), (A, 18:32:02), (C, 18:33:05), (E, 18:47:12), (C, 18:48:20), (H, 19:15:49), (C, 19:51:32) >\}$
En considérant $\Delta t = 30$ minutes, largement utilisé comme un seuil temporel standard 8 nous obtenons les trois visites suivantes :

1. $v_1 = \{u, 16:09:10, <(A, 16:09:10), (B, 16:09:43), (C, 16:12:02) >\}$,
2. $v_2 = \{u, 18:32:02, <(A, 18:32:02), (C, 18:33:05), (E, 18:47:12), (C, 18:48:20), (H, 19:15:49) >\}$ et
3. $v_3 = \{u, 19:51:32, <(C, 19:51:32) >\}$.

On a donc la session $S = \{v_1, v_2, v_3\}$

Processus du Web Usage Mining

Identification des visites et sessions

Pour identifier les sessions on se sert des variations des temps. Nous avons trois cas :

1. **h1** : la durée totale de la session ne doit pas dépasser un seuil $\theta.(t - t_0 \leqslant \theta)$
2. **h2** : Le temps total passé sur une page ne doit pas dépasser un seuil $\beta.(t_2 - t_1 \leqslant \beta)$
3. **h-ref** : une requête q est ajouté à la session construite S si le référent pour q a été précédemment invoqué dans S. (q peut appartenir à plusieurs sessions ouvertes).

Processus du Web Usage Mining

Identification des visites et sessions

Exemple d'indentification de sessions :

- h1

User 1

Time	IP	URL	Ref
0:01	1.2.3.4	A	-
0:09	1.2.3.4	B	A
0:19	1.2.3.4	C	A
0:25	1.2.3.4	E	C
1:15	1.2.3.4	A	-
1:26	1.2.3.4	F	C
1:30	1.2.3.4	B	A
1:36	1.2.3.4	D	B

Session 1

0:01	1.2.3.4	A	-
0:09	1.2.3.4	B	A
0:19	1.2.3.4	C	A
0:25	1.2.3.4	E	C

Session 2

1:15	1.2.3.4	A	-
1:26	1.2.3.4	F	C
1:30	1.2.3.4	B	A
1:36	1.2.3.4	D	B

- href

User 1

Time	IP	URL	Ref
0:01	1.2.3.4	A	-
0:09	1.2.3.4	B	A
0:19	1.2.3.4	C	A
0:25	1.2.3.4	E	C
1:15	1.2.3.4	A	-
1:26	1.2.3.4	F	C
1:30	1.2.3.4	B	A
1:36	1.2.3.4	D	B

Session 1

0:01	1.2.3.4	A	-
0:09	1.2.3.4	B	A
0:19	1.2.3.4	C	A
0:25	1.2.3.4	E	C
1:26	1.2.3.4	F	C

Session 2

1:15	1.2.3.4	A	-
1:30	1.2.3.4	B	A
1:36	1.2.3.4	D	B

Exercice

En se servant de l'extrait du fichier log ci-dessous, identifier les différentes sessions en utilisant la règle h_2, prendre le seuil $\beta = 10$.

Time	IP	URL	Ref
0 :01	1.2.3.4	A	-
0 :09	1.2.3.4	B	A
0 :19	1.2.3.4	C	A
0 :25	1.2.3.4	E	C
1 :15	1.2.3.4	A	-
1 :26	1.2.3.4	F	C
1 :30	1.2.3.4	B	A
1 :36	1.2.3.4	D	B

Processus du Web Usage Mining

Identification de l'épisode

Il existe trois méthodes pour identifier les épisodes :

1. la référence-avant maximale (MF - "Maximal Forward")
2. le typage des pages
3. la longueur de la référence

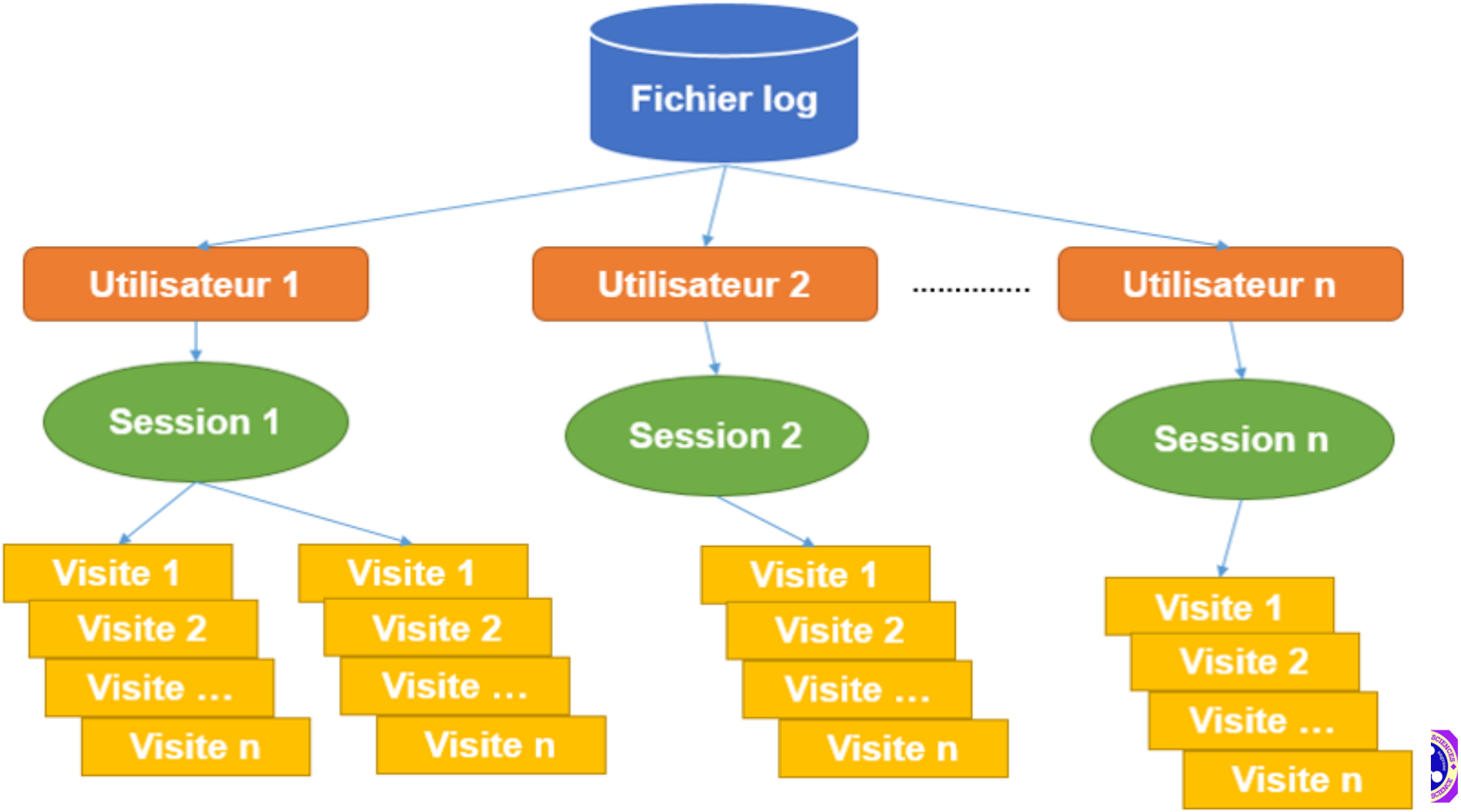
Dans la méthode MF, les auteurs ne considèrent pas une deuxième fois les pages qui ont été traversées par l'utilisateur dans sa visite. En utilisant cette méthode, la visite v_2 aura la forme : $v2 = \{u, 18:32:02, <(A, 18:32:02), (C, 18:33:05), (E, 18:47:12), (H, 19:15:49) >\}$. Cette méthode a un désavantage dans le fait que, pour certaines classes d'applications, il est important de prédire même ces types de référence en arrière.

Processus du Web Usage Mining

Identification de l'épisode

Les méthodes de typage de pages et la longueur de la référence, sont basées sur la classification de pages. Les pages sont classifiées en "type média" ou en "type auxiliaire". La différence entre les deux méthodes consiste dans l'algorithme de classification. Cet algorithme est basé sur les données d'usage pour la méthode longueur de référence et sur le contenu de la page pour la méthode typage de page.
Même si, avec ces deux méthodes, l'auteur obtient de meilleures résultats qu'avec la méthode MF, les deux se basent sur des heuristiques pour déterminer la définition sémantique du type de la page

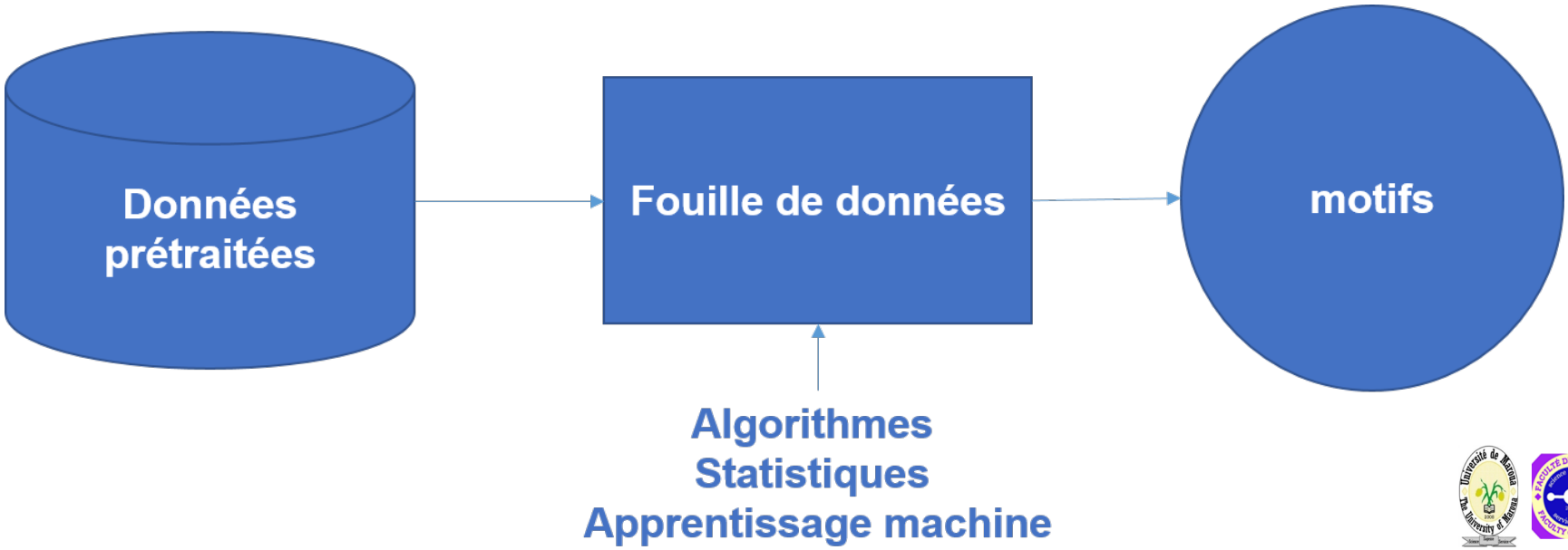

Processus du Web Usage Mining



Processus du Web Usage Mining

Extractions d'informations

- La fouille de données permet d'appliquer une des techniques d'extraction pour la découverte des motifs.
- Cette découverte s'appuie sur des méthodes et des algorithmes développés à partir de plusieurs domaines tels que les statistiques, le Data Mining, l'apprentissage machine et la reconnaissance des formes.



Processus du Web Usage Mining

Analyse des motifs extraits

- Cette analyse nécessite le recours à un ensemble d'outils pour ne garder que les résultats les plus pertinents et les plus significatifs.
- Elle est considérée comme une étape importante du processus d'extraction, car une fois les motifs trouvés, il faut être en mesure de bien sélectionner les motifs intéressants et de pouvoir les valider.