Présentation
SSDM : Semantically Similar Data Miner

Guillaume Calas <calas_g@epita.fr>
Henri-François Chadeisson <chadei_hepita.fr>

EPITA SCIA 2009

16 Juillet 2008

Plan

. . .


BOIRE OU FOUILLER, FAUT CHOISIR !
MOI, C'EST DÉCIDÉ, JE FAIS LES DEUX !
HIPS!
CMPB
www.cmpb.net

Plan

Plan

Le *data mining* sans les petites roues (1/2)

Définition

Analyse portant sur un ensemble de données dans lequel aucune des données ou valeur prise individuellement n'a de valeur particulière et dont aucune n'est la cible, l'objet de l'analyse.

Le *data mining* sans les petites roues (1/2)

Définition

Analyse portant sur un ensemble de données dans lequel aucune des données ou valeur prise individuellement n'a de valeur particulière et dont aucune n'est la cible, l'objet de l'analyse.

Objectif global

L'objectif global étant la découverte de corrélations intéressantes (inconnues jusqu'alors) entre les données à partir d'une très grande quantité de données.

Le *data mining* sans les petites roues (2/2)

Quelques applications

- dégager des groupes homogènes à partir d'un ensemble d'individus

Le *data mining* sans les petites roues (2/2)

Quelques applications

- dégager des groupes homogènes à partir d'un ensemble d'individus
- construire des normes de comportement

Le *data mining* sans les petites roues (2/2)

Quelques applications

- dégager des groupes homogènes à partir d'un ensemble d'individus
- construire des normes de comportement
  - compression d'informations

Le *data mining* sans les petites roues (2/2)

Quelques applications

- dégager des groupes homogènes à partir d'un ensemble d'individus
- construire des normes de comportement
  - compression d'informations
  - détection de déviation par rapport à ces normes

Le *data mining* sans les petites roues (2/2)

Quelques applications

- dégager des groupes homogènes à partir d'un ensemble d'individus
- construire des normes de comportement
  - compression d'informations
  - détection de déviation par rapport à ces normes

Techniques utilisées

- **Réseau de neurones :** carte de KOHONEN, cartes auto-adaptatives, etc..

Le *data mining* sans les petites roues (2/2)

Quelques applications

- dégager des groupes homogènes à partir d'un ensemble d'individus
- construire des normes de comportement
  - compression d'informations
  - détection de déviation par rapport à ces normes

Techniques utilisées

- **Réseau de neurones :** carte de KOHONEN, cartes auto-adaptatives, etc..
- **Méthodes statistiques :** K-means, ACP, NNS, etc..

Le *data mining* sans les petites roues (2/2)

Quelques applications

- dégager des groupes homogènes à partir d'un ensemble d'individus
- construire des normes de comportement
  - compression d'informations
  - détection de déviation par rapport à ces normes

Techniques utilisées

- **Réseau de neurones :** carte de KOHONEN, cartes auto-adaptatives, etc..
- **Méthodes statistiques :** K-means, ACP, NNS, etc..
- **Data mining :** *apriori*, SSDM, Carma, etc..

Plan

Ensembles flous et *data mining* (1/2)

Origines

- 1965 - Lofti Asker ZADEH
- Objectif : Représenter mathématiquement l'imprécision relative à certaines classes d'objets
- Sert de fondement à la logique floue

Utilisation

- Modéliser la représentation humaine des connaissances
- Améliorer les performances des systèmes de décision qui utilisent cette modélisation
- Modéliser l'incertitude et l'imprécision

Ensembles flous et *data mining* (2/2)

Algorithmes utilisant les ensembles flous

- *Lee and Lee-Kwang's algorithm*[6] ;
- *Kuok, Fu and Wong's algorithm*[5] ;
- *F-APACS*[2] ;
- *FARM*[1] ;
- *FTDA*[4].

FIG.: Algorithmes travaillant sur des données quantitatives

Plan

SSDM (1/3)

Description

- SSDM : *Semantically Similar Data Miner* [3]
- Objectif : Mener une étude sémantique sur les données traitées afin d'améliorer et d'augmenter les relations mises en évidence

Principe

- Utiliser la logique floue afin de mettre en évidence des liens sémantiques entre différents éléments.
- Algo basé sur la génération de règles associatives entre *itemsets*.
- On définit les variables suivantes :
  - *support* : $\frac{\text{Transactions comprenant X et Y}}{\text{Nombre total de transactions}}$
  - *confidence* : $\frac{\text{Transactions comprenant X et Y}}{\text{Nombre de transactions comprenant X}}$

SSDM (2/3) - Support / Confidence : Exemples

Attribut 1	*Attribut 2*
chair	table
sofa	desk
chair	desk
chair	table

$chair \Rightarrow table(support = 50\%, confidence = 67\%)$
$sofa \Rightarrow desk(support = 25\%, confidence = 100\%)$
$chair \Rightarrow desk(support = 25\%, confidence = 33\%)$

SSDM (3/3)

Sélection de règles

- Support $(X \Rightarrow Y) > 50\%$
- Confidence $(X \Rightarrow Y) > 60\%$

$chair \Rightarrow table(support = 50\%, confidence = 67\%)$

~~$sofa \Rightarrow desk(support = 25\%, confidence = 100\%)$~~

~~$chair \Rightarrow desk(support = 25\%, confidence = 33\%)$~~

SSDM (3/3)

Sélection de règles

- Support $(X \Rightarrow Y) > 50\%$
- Confidence $(X \Rightarrow Y) > 60\%$

$chair \Rightarrow table(support = 50\%, confidence = 67\%)$
~~$sofa \Rightarrow desk(support = 25\%, confidence = 100\%)$~~
~~$chair \Rightarrow desk(support = 25\%, confidence = 33\%)$~~

But de l'algorithme

- Mots "Table" et "Desk" différents
- Concepts "Table" et "Desk" proches

Mettre en évidence ces relations conceptuelle par une approche sémantique, afin de permettre de trouver de nouvelles relations.

Plan

Plan

Description et Initialisation

Description

1. *Data scanning* (préparation des données)
2. Remplissage de la matrice de similarités
3. Identification des éléments similaires, cycle de similarités
4. Génération des candidats
5. Pondération des ensembles candidats
6. Évaluation des candidats
7. Génération des règles associatives

Description et Initialisation

Description

1. *Data scanning* (préparation des données)
2. Remplissage de la matrice de similarités
3. Identification des éléments similaires, cycle de similarités
4. Génération des candidats
5. Pondération des ensembles candidats
6. Évaluation des candidats
7. Génération des règles associatives

Initialisation

- 3 valeurs seuils (*minsup*, *minconf*, *minsim*)
- matrices de similarité

Plan

Base de travail

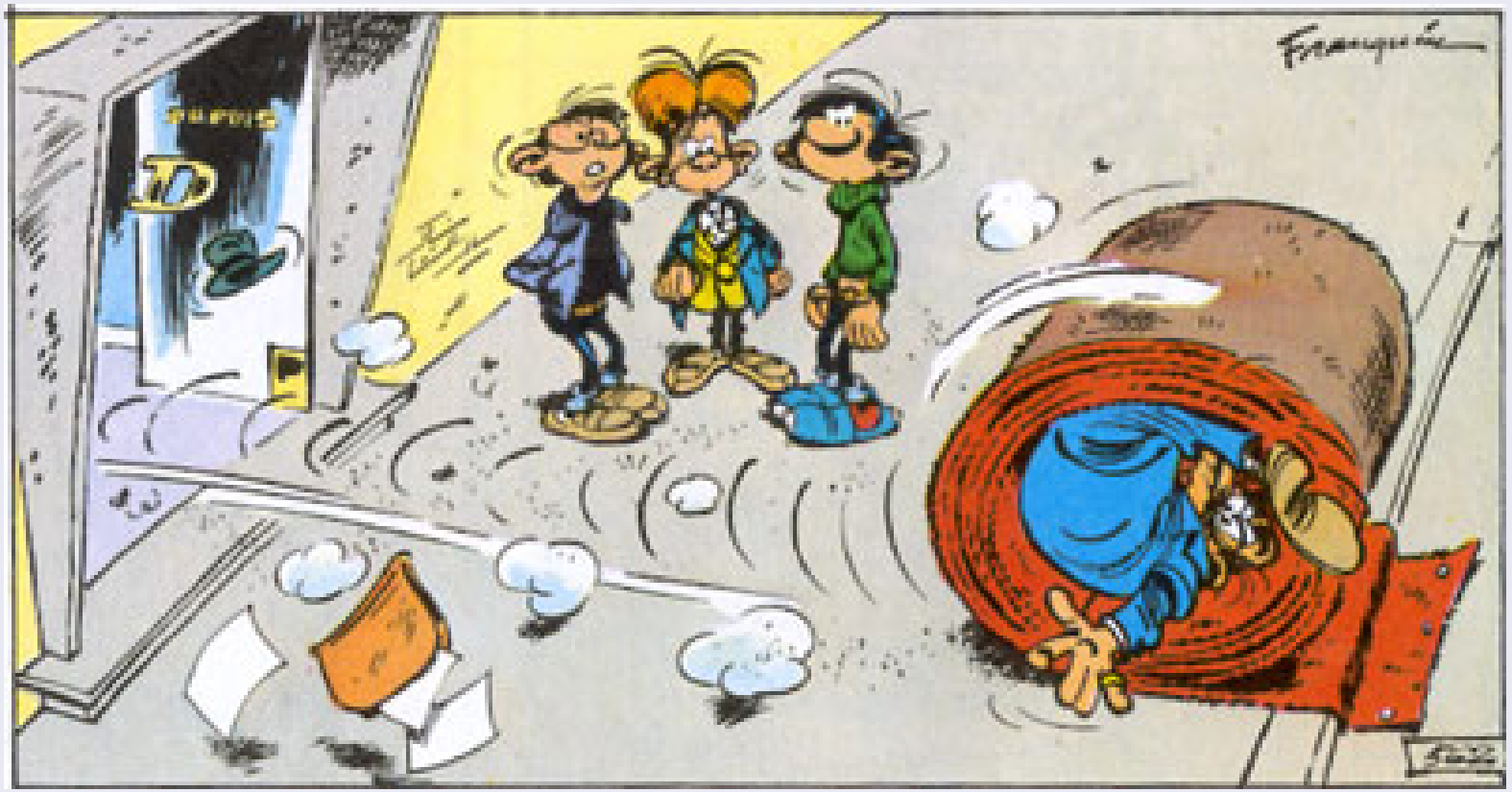
id	Attribut 1	Attribut 2	Attribut 3
10	chair	table	wardrobe
20	sofa	desk	cupboard
30	seat	table	wardrobe
40	sofa	desk	cupboard
50	chair	board	wardrobe
60	chair	board	cupboard
70	chair	desk	cupboard
80	seat	board	cabinet
90	chair	desk	cabinet
100	sofa	desk	cupboard

- Minimal support ($minsup$) $= 0.45$
- Minimal confidence ($minconf$) $= 0.3$
- Minimal similarity ($minsim$) $= 0.8$

Plan

0 - Comment ça se passe ?

Non, on déroule ...



1 - Data scanning

Attribuer un domaine à chaque élément (Pré-tri).
Lorsque les données sont stockées en BDD, on peut définir un domaine par champs.

Éléments	Domaines
sofa, chair, seat	Domaine 1
board, desk, table	Domaine 2
cabinet, cupboard, wardrobe	Domaine 3

- *Domaine 1* contient les éléments sur lesquels on peut s'asseoir.
- *Domaine 2* contient les éléments sur lesquels on peut poser quelque chose.
- *Domaine 3* contient les éléments qui peuvent contenir quelque chose.

2 - Degrés de similarité

- Déterminer pour chaque domaine des degrés de similarité entre ses éléments (Matrice de similarité).
- Peut être fait à la main, ou automatiquement.

Dom 1	*Sofa*	*Seat*	*Chair*	Dom 2	*Desk*	*Table*	*Board*
Sofa	1	0.75	0.7	*Desk*	1	0.9	0.75
Seat	0.75	1	0.6	*Table*	0.9	1	0.7
Chair	0.7	0.6	1	*Board*	0.75	0.7	1

Dom 3	*Cabinet*	*Wardrobe*	*Cupboard*
Cabinet	1	0.9	0.85
Wardrobe	0.9	1	0.8
Cupboard	0.85	0.8	1

3 - Eléments similaires - Ensembles flous

Domaine	Valeur	Relation
Domaine 2	0.9	Desk∼Table
Domaine 3	0.9	Cabinet∼Wardrobe
Domaine 3	0.85	Cabinet∼Cupboard
Domaine 3	0.8	Cupboard∼Wardrobe

FIG.: Table associative avec *minsim*= 0.8

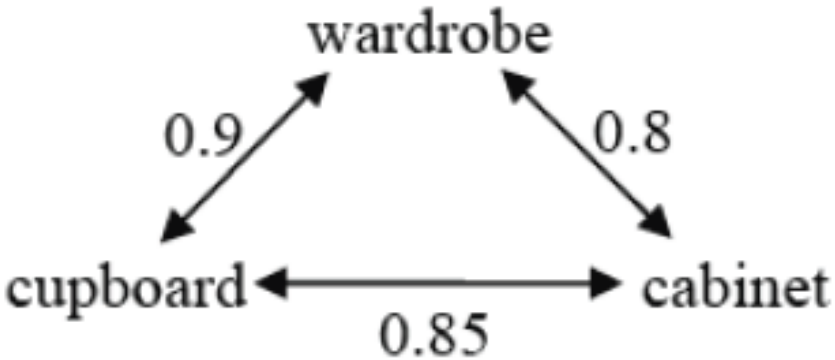


FIG.: Cycle de similarité

4 - Génération des candidats

La génération des candidats est similaire à celle de l'algorithme *Apriori* où on rajoute les éléments considérés comme similaires aux ensembles générés.

1) $L_1 = \{\text{large 1-itemsets}\}$;
2) **for** ($k = 2$; $L_{k-1} \neq \emptyset$; k++) **do begin**
3) $C_k = \text{apriori-gen}(L_{k-1})$; // New candidates
4) **forall transactions** $t \in \mathcal{D}$ **do begin**
5) $C_t = \text{subset}(C_k, t)$; // Candidates contained in t
6) **forall candidates** $c \in C_t$ **do**
7) c.count++;
8) **end**
9) $L_k = \{c \in C_k \mid c.\text{count} \geq \text{minsup}\}$
10) **end**
11) Answer $= \bigcup_k L_k$;

FIG.: Apriori algorithm

5 - Poids des ensembles candidats (1/2)

5 - Poids des ensembles candidats (1/2)

Exemple

Somme des occurences des éléments de l'ensemble pondérées par les coefficients de similarité.

5 - Poids des ensembles candidats (1/2)

Exemple

Somme des occurences des éléments de l'ensemble pondérées par les coefficients de similarité.

	$Item_1$	$Item_2$
$Item_1$	1	0.8
$Item_2$	0.8	1

On dispose d'un *ensemble* contenant 2 fois l'$Item_1$ et 1 fois l'$Item_2$.

5 - Poids des ensembles candidats (1/2)

Exemple

Somme des occurences des éléments de l'ensemble pondérées par les coefficients de similarité.

	$Item_1$	$Item_2$
$Item_1$	1	0.8
$Item_2$	0.8	1

On dispose d'un *ensemble* contenant 2 fois l'$Item_1$ et 1 fois l'$Item_2$.

Pour 2 éléments

$poids(Item_1) + poids(Item_2) \times sim(Item_1, Item_2) = 2.8$

5 - Poids des ensembles candidats (1/2)

Exemple

Somme des occurences des éléments de l'ensemble pondérées par les coefficients de similarité.

	$Item_1$	$Item_2$
$Item_1$	1	0.8
$Item_2$	0.8	1

On dispose d'un *ensemble* contenant 2 fois l'$Item_1$ et 1 fois l'$Item_2$.

Pour 2 éléments

$poids(Item_1) + poids(Item_2) \times sim(Item_1, Item_2) = 2.8$
$poids(Item_1) \times sim(Item_1, Item_2) + poids(Item_2) = 2.6$

5 - Poids des ensembles candidats (1/2)

Exemple

Somme des occurences des éléments de l'ensemble pondérées par les coefficients de similarité.

	$Item_1$	$Item_2$
$Item_1$	1	0.8
$Item_2$	0.8	1

On dispose d'un *ensemble* contenant 2 fois l'$Item_1$ et 1 fois l'$Item_2$.

Pour 2 éléments

$$poids(Item_1) + poids(Item_2) \times sim(Item_1, Item_2) = 2.8$$
$$poids(Item_1) \times sim(Item_1, Item_2) + poids(Item_2) = 2.6$$
$$poids = \frac{[poids(Item_1)+poids(Item_2)][1+sim(Item_1,Item_2)]}{2} = 2.7$$

5 - Poids des ensembles candidats (2/2)

Pour plus de 2 éléments

$$f = min\left(sim\left(Item_1, Item_2\right), ..., sim\left(Item_{n-1}, Item_n\right)\right)$$
$$poids = \left[\sum_{i=1}^{n} poids\left(Item_i\right)\right]\left[\frac{1+f}{2}\right]$$

6 - Évaluation des candidats

Évaluation des candidats

Les ensembles d'éléments dont le support est inférieur à *minsup* sont écartés car non suffisament fréquents.

$$Support = \frac{poids(\text{Ensemble d'éléments})}{\text{Nombre d'entrées dans la base de données}}$$

7 - Génération des règles associatives

Rappel

Une règle d'association est définie par :

$$Antecedent \Rightarrow Consequent$$

Génération des règles

L'ensemble des règles possibles entre chaque ensemble sont générées et seules les règles dont la valeur de confidence est supérieure à *minconf* sont conservées.

$$Confidence = \frac{Support(Regle)}{Support(Antecedent)}$$

Plan

Plan

Description du test - Comparaison SSDM / Apriori

Datasets : Personnes atteintes du SIDA au Brésil

Dataset AIDS1

3324 entrées Champs : Région (22 régions), Sexe, Age, Status (single / married / widow(er) / coupled / separated / divorced / did not answer)

Dataset AIDS2

3324 entrées Champs :

- Religion (catholic / protestant / pentecostal / spiritism / afro-brazilian / none / did not answer / other)
- Do you read newspaper ? (yes, diary / yes, almost every day / yes, once in a week / yes, once in a while / yes, rarely / never / did not answer)
- Do you watch TV ?

Dataset 1 - Initialisation

$Attribute_1$	$Attribute_2$	Degré de similarité
Single	Widow(er)	0.7
Married	Coupled	0.8
Separated	Divorced	0.9

- *minsup* : 0.2
- *minconf* : 0.4
- *minsim* : 0.7

Dataset 1 - Résultats

Apriori

married → female sup=0.23495789 conf=0.54922646
female → married sup=0.23495789 conf=0.4256131

Dataset 1 - Résultats

Apriori

married → female sup=0.23495789 conf=0.54922646
female → married sup=0.23495789 conf=0.4256131

SSDM

coupled∼married→male sup=0.23420578 conf=0.46331015
male→coupled∼married sup=0.23420578 conf=0.5228341
married→female sup=0.23495789 conf=0.54922646
female→married sup=0.23495789 conf=0.4256131
coupled∼married→female sup=0.27129963 conf=0.5366899
female→coupled∼married sup=0.27129963 conf=0.49144414

Dataset 2 - Initialisation

$Attribute_1$	$Attribute_2$	Degré de similarité
(N) Yes, diary	(N)Yes, Almost every day	0.8
(N) Yes, rarely	(N) Once in a while	0.6
(T) Yes, diary	(T)Yes, Almost every day	0.8
(T) Yes, rarely	(T) Once in a while	0.6

- *minsup* : 0.2
- *minconf* : 0.7
- *minsim* : 0.6

Dataset 2 - Résultats

Apriori

(T) Yes, diary → Protestant sup=0.5744222 conf=0.77312315
Protestant → (T) Yes, diary sup=0.5744222 conf=0.768343
(N) Never → Protestant sup=0.24807397 conf=0.76520914

Dataset 2 - Résultats

Apriori

(T) Yes, diary → Protestant sup=0.5744222 conf=0.77312315
Protestant → (T) Yes, diary sup=0.5744222 conf=0.768343
(N) Never → Protestant sup=0.24807397 conf=0.76520914

SSDM

(N) Yes, rarely∼(N) Once in a while → (T) Yes, diary sup=0.25565487 conf=0.78620166
(N) Yes, rarely∼(N) Once in a while → Protestant sup=0.24086288 conf=0.74071264
(T) Yes, diary → Protestant sup=0.5744222 conf=0.77312315
Protestant → (T) Yes, diary sup=0.5744222 conf=0.768343
(N) Never → Protestant sup=0.24807397 conf=0.76520914

Plan

Critiques

Autocritique

- Algorithme plus lent que *Apriori*
- Édition manuelle des matrices de similarité

Notre avis

- Intérêt de *minsim* ?
- *Datasets* de tests trop petits
- Pas d'étude sur le temps d'exécution

Bons points

- Analyse sémantique des données
- On génère bien plus de règles
- Acquisition de nouvelles connaissances
- Utilisation de logique floue $\Rightarrow$ Approche plus fidèle au raisonnement humain
- Les nouvelles règles générées sont pertinentes

Conclusion

Bilan

- Mise en évidence de nouvelles associations pertinentes
- $\Rightarrow$ Objectif initial atteint

Améliorations

- Améliorer les performances de l'algorithme en modifiant les structures de données utilisées
- Recherche de techniques de définition de domaines automatisées
- *Data cleaning*
- Définition automatisée des matrices de similarité (regain d'intérêt pour *minsim*)

Questions

We want YOUR questions !

Questions

We want YOUR questions !

Vous pouvez détacher vos ceintures. . .

. . . et merci de votre attention. . .

Bibliographie I


W. Au and K. Chan.
Farm : A data mining system for discovering fuzzy association rules.
In Proc. of the 8th IEEE Int'l Conf. on Fuzzy Systems, pages 1217–1222, Seoul, Korea, 1999.


K. C. C. Chan and W.-H. Au.
An effective algorithm for mining interesting quantitative association rules.
In SAC, pages 88–90, 1997.


E. L. G. Escovar, M. Biajiz, and M. T. P. Vieira.
Ssdm : A semantically similar data mining algorithm.
In C. A. Heuser, editor, SBBD, pages 265–279. UFU, 2005.

Bibliographie II


T.-P. Hong, C.-S. Kuo, S.-C. Chi, and S.-L. Wang.
Mining fuzzy rules from quantitative data based on the aprioritid algorithm.
In SAC (1), pages 534–536, 2000.


C. M. Kuok, A. W.-C. Fu, and M. H. Wong.
Mining fuzzy association rules in databases.
SIGMOD Record, 27(1) :41–46, 1998.


J.-H. Lee and H. Lee-Kwang.
Fuzzy identification of unknown systems based on ga.
In X. Yao, J.-H. Kim, and T. Furuhashi, editors, SEAL, volume 1285 of Lecture Notes in Computer Science, pages 216–223. Springer, 1996.