Extension de l'algèbre relationnelle aux données symboliques

Tao WAN, Karine ZEITOUNI

Labo. PRISM – Université de Versailles Saint-Quentin

Plan

- Introduction
  - Contexte et motivations
- État de l'art
  - Algèbres étendues
  - Similarité des données symboliques
- Extension de l'algèbre relationnelle au symbolique
  - Opérateurs spécifiques

Motivation

Applications

- De nombreuses applications nécessitent la gestion de données non atomiques
  - Ex: Mesures de vitesse ou de pollution forment des distributions liées à des intervalles de temps et des localisations géographiques
- Il est nécessaire de gérer, d'explorer et d'analyser ces représentations complexes pour comprendre le phénomène étudié.

Introduction aux données symboliques (1)

- **Idée d'origine**

  Agréger et résumer les données, à l'aide de concepts de plus haut niveau afin de mieux les appréhender et d'en extraire de nouvelles connaissances.

- **Avantages**
  - Description intelligible et d'une taille plus maniable.
  - Pouvoir contenir de la variation interne et des structures compliqués
- **Applications**
  - Exprimer des résumés en assurant la confidentialité des données originales. Ex: Recensement de la population: regrouper des îlots sans perte d'information
  - Exprimer des données imprécises. Ex: données imprécises natives (poids, valeurs)
  - Exprimer la contenu des objets multimédias. Ex: histogramme des couleurs

Introduction aux données symboliques (2)

Un exemple de génération d'une table de données symbolique:

Id	Couleur	Matériel
1	Rouge	Fer
1	Bleu	Cuivre
1	Rouge	Cuivre
2	Bleu	Cuivre
3	Vert	Bronze
3	Bleu	Fer

⋈

Id	Température
1	30°
2	45°
2	25°
3	39°

Select Id, MostFrequent(Couleur), MostFrequent(Matériel) Min(Température)

From Table 1

Group by Id

Id	Couleur	Matériel	Température
1	Rouge	Cuivre	30°
2	Bleu	Cuivre	25°
3	Vert	Bronze	39°

Id	Couleur	Matériel	Température
1	Rouge	Fer	30°
1	Bleu	Cuivre	30°
1	Rouge	Cuivre	30°
2	Bleu	Cuivre	45°
2	Bleu	Cuivre	25°
3	Vert	Bronze	39°
3	Bleu	Fer	39°

Table 1

Table symbolique obtenue :

Id	Couleur	Matériel	Température
1	Rouge, Bleu	Fer (33%), Cuivre(67%)	30°
2	Bleu	Cuivre (100%)	[25°, 45°]
3	Vert, Bleu	Bronze (50%), Fer (50%)	39°

Introduction des données symboliques (3)

Les variables symboliques peuvent être :

1 . Intervalles

Par exemple Période_de_Mesure = [16, 20]

2 . Multi-valuées

Par exemple Voisinage_Capteur = {Usine, Chaufferie}

3 . Multivaluées avec pondération

Par exemple Pollution_CO = {Faible (10%), Moyenne (50%), Forte (40%)}

Objets symboliques

Un concept est défini par
- Intension : les propriétés caractéristiques d'une classe des individus
- Extension : la classe des individus de la base satisfaisant ces propriétés.

Un objet symbolique est une modélisation d'un concept

Objet symbolique: est un triplet s = (a , R , d) où
- d est une description de domaine D (valeur).
- R une relation sur domaine D permettant de comparer d à une autre description de D. (comparateur du prédicat)
- a est une fonction permettant d'évaluer le résultat de la comparaison (à l'aide de R) de la description d'un individu de Ω par rapport à la description de données d. (binaire ou modale)

Un objet symbolique ⇔ requête

Deux type d'objets symboliques

Si T est une table symbolique, t est un tuple , $t \in T$ et t_i est la valeur symbolique d'attribut i.

- **Les objets symboliques booléens :**

  c'est le cas où $a = [t(T)\ R\ d] = \wedge\ [t_i(T)\ R_i\ d_i]: E \rightarrow \{$ vrai , faux $\}$

  Ex: $d_i = \{$ rouge, bleu, jaune$\}$, $t_i(T) = \{$ rouge, jaune $\}$

  $R_i = \subseteq$, impliquent $a_i(\omega) = [t_i(T) \subseteq d_i] =$ vrai

- **Les objets objets symboliques modaux :**

  c'est le cas où $a = [t(T)\ R\ d] = \wedge\ [t_i(T)\ R_i\ d_i]: E \rightarrow [\ 0\ ,\ 1\ ]$

  Ex: $d_i = \{$ (0.2)rouge , (0.3)bleu , (0.1)jaune $\}$,

  $t_i(T) = \{$(0.4) rouge, (0.6)jaune $\}$,

  si R_i est le produit scalaire,

  on a donc: $a_i(\omega) = [t_i(T)\ R_i\ d_i] = 0.2*0.4 + 0.3*0 + 0.1*0.1$

Extension d'un objet symbolique

Soit T une table de type symbolique et t un tuple, $t \in T$

- Le cas booléen

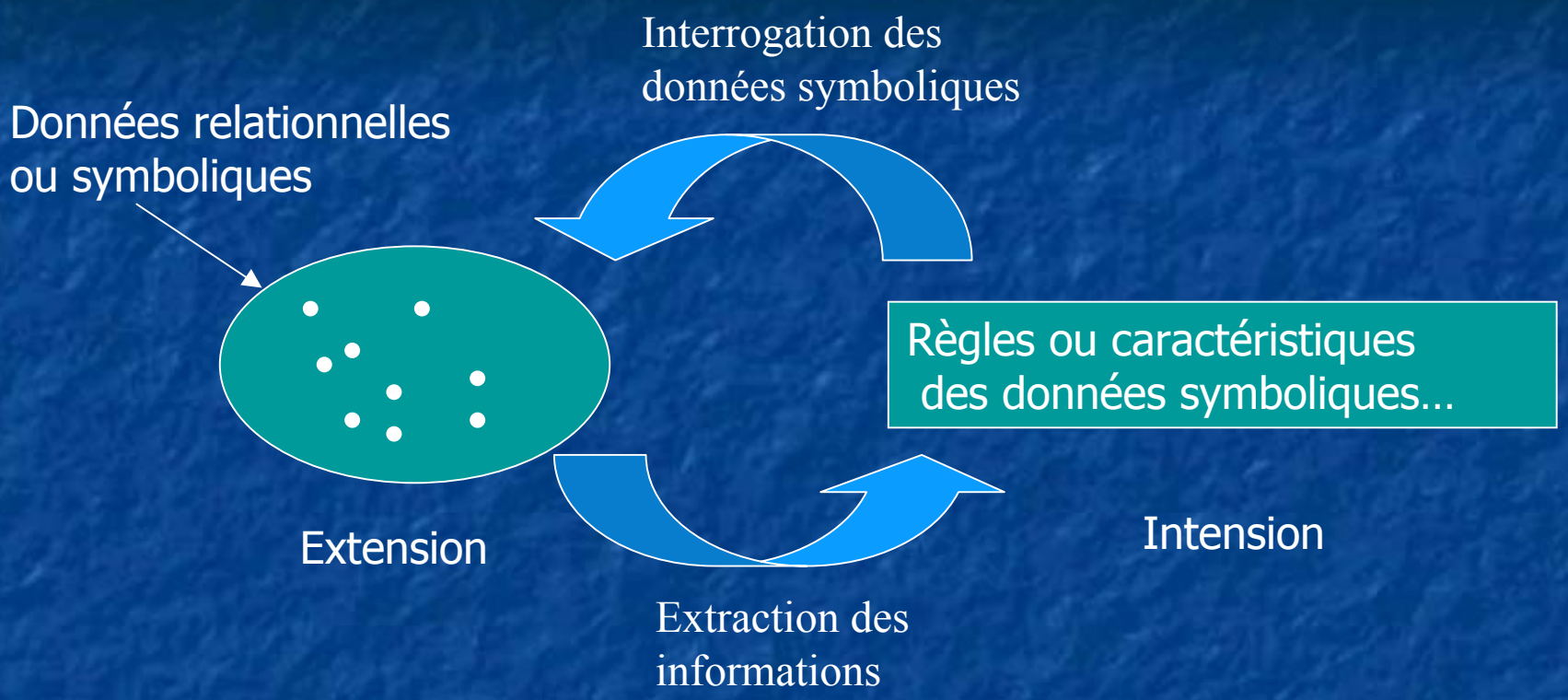
  EXT(a) = {$t \in$ T / a(t) = TRUE}.

- Le cas modal

  EXT_α (S)= $EXTENT_\alpha$ (a) = {$t \in$ T / a(t) $\geq \alpha$}. (α est un seuil)

L'extension d'un objet symbolique $\Leftrightarrow$ résultat de la requête

Problématique



- Problème: Comment interroger des données symboliques, telles que de type multi-valuées avec pondération ?
- N'est pas encore étendu au cas des bases de données symboliques.

Extension des requêtes algébriques aux données mal connues de types possibilistes

- Extension de l'algèbre relationnelle aux requêtes possibilistes adressées aux bases de données mal connues et de types possibilistes.[Bosc et al. 02]
  - Données mal connues présentées par une distribution de possibilités.
    - Il doit exister au moins une valeur de possibilité égale à 1.
- Requêtes possibilistes de la forme « Dans quelle mesure est-il possible que le n-uplet appartienne à la réponse à la requête Q » où Q est une requête relationnelle usuelle.

- La manière de présentation des données ne corresponds pas à notre cas.

- Requêtes possibiliste consistent à calculer le degré de possibilité qu'un n-uplet appartienne à la réponse.

Extension des requêtes algébriques aux données multimédia

- Données multimédia

  Ex: Attributs décrivant la contenu des images contiennent des données complexes: histogramme de couleurs, vecteur de texture…

- Langage de requête floues pour interroger des données multimédia [Ciaccia et al. 01]
  - Données multimédia décrites par des données pondérées
  - « Similarity Algebra » interroge ces données avec un matching d'imprécision.
  - Extension de requêtes
- Exemple des requête floue en utilisant la similarité
  - "Finding paintings with a texture similar to a given input texture vector"

Extension de l'algèbre relationnelle aux données symboliques

- Une requête (conditions sur des tuples)
  - Décrite par un ensemble de prédicats combinés dans une formule f selon la syntaxe $f ::= p \mid f \wedge f \mid f \vee f \mid \neg f \mid (f)$ où f est une formule et p est un prédicat
- Un tuple t
  - un prédicat flou étend le prédicat booléen avec une mesure de résultat entre 0 et 1.
    - Un prédicat flou est de la forme:
      - $A \approx v$ (ou v est une valeur constante) ou $A_1 \approx A_2$ où A_1 et A_2 sont dans le même domaine.
    - Avec s (p_i, t) un degré de satisfaction (score) d'un prédicat pi pour le tuple t donné. ($s(p_i, t) \in [0, 1]$)
  - le calcul de s (pi, t) est basé sur la similarité des données symboliques
  - $s\,(f(p_1, \ldots, p_n), t) = s_f\,(s(p_1, t), \ldots, s(p_n, t))$ s_f est la fonction de score [Ciaccia et al. 01]

	FS
$s(p_1 \wedge p_2, t)$	$\text{Min}(s(p_1, t), s(p_2, t))$
$s(p_1 \vee p_2, t)$	$\text{Max}(s(p_1, t), s(p_2, t))$
$s(\neg p, t)$	$1 - s(p, t)$

Traduction des opérations logiques And($\wedge$), Or($\vee$) et Not($\neg$)

dans FS (Fuzzy Standard)

Sélection et projection symboliques

- Degré de satisfaction d'une sélection:

  Degré de satisfaction *s* sur un attribut *i* = la probabilité des valeurs superposés [Bock 01] :

  - *Si* t_i est la valeur d'un attribut *i* d'un tuple symbolique *t* et d_i une description, l'évaluation $s(p_i, t) = t_i \cap d_i / t_i \cup d_i$.

  Ex: t_i = {(0.4) rouge , (0.5) bleu , (0.1) jaune}, d_1 = {rouge = 0.4} **et** d_2 = {jaune = 0.3},
  $s\,(f(p_1 \wedge p_2), t_i) = s_f\,(s(p_1, t_i), s(p_2, t_i))$ = min(s(p_1, t_i), s(p_2, t_i)) = min (0.4/0.4, 0.1/0.3) = 1/3

- Dans le cas de projection pour éliminer les tuples en double:

  2 fonctions de comparaison de dissimilarité [Malerba et al. 02] :

  $a = [y_1 \in A_1] \wedge [y_2 \in A_2] \wedge \ldots \wedge [y_n \in A_n]$, $b = [y_1 \in B_1] \wedge [y_2 \in B_2] \wedge \ldots \wedge [y_n \in B_n]$

  - *Weighted Minkowski's metric:*
    $m(A_k, B_k) = \sum_{y \in Yk} |p(y_k) - q(y_k)|$

    $$d_p(a,b) = \sqrt[p]{\sum_{k=1}^{n} [c_k m(A_k, B_k)]^p}$$

  - *Aggregated dissimarity:*

    $$d_p(a,b) = \frac{\prod_{i=1}^{n} (\sqrt[p]{2} - \sqrt[p]{\sum_{y_i} |p(y_i) - q(y_i)|^p})}{(\sqrt[p]{2})^n} = 1 - \frac{\prod_{i=1}^{n} (\sqrt[p]{2} - \sqrt[p]{L_p})}{(\sqrt[p]{2})^n}$$

Attention: Ici, on ne s'intéresse qu'aux données symboliques de type distribution

Exemple de sélection

- Sélection

  Opération sur une table symbolique produisant une nouvelle table symbolique *de même schéma*, mais comportant les seuls tuples qui vérifient la condition précisée en argument.

- Exemple:

  Select * from table 1
  Where (Couleur = {rouge} or Couleur = {bleu}) and (Matériel = {Cuivre >= 0.5} or Matériel = { Bronze = 1}) and (températeur > 30° or températeur < 80°)
  With s_f >= 50%

Id	Couleur	Matériel	Température
1	Rouge, Bleu	Fer(0.2), Cuivre(0.8)	30°
2	Bleu	Cuivre(1)	25°
3	Vert, Bleu	Bronze(0.5), Fer(0.5)	39°
4	Rouge	Bronze(1)	80°

σ (Couleur = {rouge} or Couleur = {bleu}) and (Matériel = {Cuivre >= 0.5} or Matériel = { Bronze = 1}) and (températeur > 30° or températeur < 80°)with sf >= 50%

Id	Couleur	Matériel	Température
1	Rouge, Bleu	Fer(0.2), Cuivre(0.8)	30°
3	Vert, Bleu	Bronze(0.5), Fer(0.5)	39°

Exemple de projection

- Projection

  Opération sur une table symbolique consistant à composer une nouvelle table symbolique en enlevant à la table initiale tous les attributs non mentionnés en opérandes (aussi bien au biveau du schéma que des tuples) et en éliminant les tuples en double qui sont conservés une seul fois.

- possibilités en symbolique
  - **Supprimer les doublons strictement identiques**
  - **Ou bien similaires** (seuil de similarité défini par l'utilisateur**) !**
    garder le max ou encore agréger par somme ??

Id	Couleur	Matériel	Température
1	Rouge, Bleu	Fer(0.2), Cuivre(0.8)	30°
2	Bleu	Cuivre(1)	25°
3	Rouge, Bleu	Bronze(0.5), Fer(0.5)	39°
4	Rouge	Bronze(1)	80°

$\pi_{\text{Matériel}}$

Couleur
Rouge, Bleu
Rouge Bleu

Exemple de jointure

- Jointure par similarité (deux solutions)
  - Retourner les deux colonnes sur lesquelles porte le critère de jointure : pour ne pas perdre d'information.
  - Explicitement donner une combinaison des résultats dans une forme demandée par utilisateur

T_1

Id	A	B
1		
2		

T_2

id	A	C
1		
2		

$T_1.A \approx T_2.A$

Id	T1.A	T2.A	B	C
T1.1				
T1.2				
T2.1				

Conclusion

- Premiers pas de résolution d'objets symboliques par des requêtes:
  - Impliquent d'étendre les types et les opérateurs algébriques
  - Implémentation en cours par un UDT sous Oracle 9i
- Et après ...
  - Non tranché : projection, calculs d'attributs et agrégats ?
  - Les tables symboliques permettent-elles de supporter la notion de concept de haut niveau et de gestion des connaissances symboliques induites ?
  - Est-ce simplement un agrégat spécifique.

Bibliographie

[Bosc 02] P. Bosc, L. Duval, O. Pivert About Selections and Joins in Possibilistic Queries Addressed to Possibilistic Databases. In *Proc DEXA2002*.

[Laurent 02] A. Laurent Bases de données multidimensionnelles floues et leur utilisation pour la fouille de données. *Thèse de doctorat de l'Université Paris 6*.

[Ciaccia 01] P. Ciaccia, D. Montesi, W. Penzo and A. Trombetta Fuzzy Query Languages for Multimedia Data. In *Design and Management of Multimedia Information Systems: Opptunities and Challenges, M.R, Syed editor, Idea Group Publishing, Hershey, PA, USA, 2001*

[Malerba 02] D. Malerb, F.Esposito and M. Monopoli Comparing dissimilarity measures for probabilistic symbolic objects. In *Proc Data Mining III* 2002.

[Bock 01] H.-H. Bock, E. Diday Analysis of Symbolic Data – Exploratory Methods for Extracting Statistical Complex Data

Extension des requêtes algébriques au cas des données mal connus de types possibilistes

- La théorie de possibiliste[ZAD 78]

  Un modèle ordinal de l'incertain dans lequel l'imprécision est représentée au moyen d'une relation de préférence définissant un ordre total sur les situations possibles. L'idée de ce modèle est de contraindre les valeurs que pet prendre une variable par un ensemble flou normalisé (i.e., où au moins un élément appartient complètement à l'ensemble)

- Une distribution de possibilistes

- Fs

Extension des requêtes algébriques aux données mal connues de types possibilistes

#i	T-a	Date	Lieu
i_1	a_1	$1/d_3 + 0.7/d_1$	l_1
i_2	$1/a_3 + 0.3/a_4$	d_1	l_2

Elle présente 8 mondes où chaque monde est une base de données précise avec chacun un degré de possibilité

$\Pi = 1$

#i	T-a	Date	Lieu
i_1	a_1	d_3	l_1
i_2	a_3	d_1	l_1

$\Pi = 0.7$

#i	T-a	Date	Lieu
i_1	a_1	d_1	l_1
i_2	a_3	d_1	l_1

$\Pi = 0.3$

#i	T-a	Date	Lieu
i_1	a_1	d_1	l_1
i_2	a_4	d_1	l_1

$\Pi = 0.3$

#i	T-a	Date	Lieu
i_1	a_1	d_3	l_1
i_2	a_4	d_1	l_1

$\Pi = 1$

#i	T-a	Date	Lieu
i_1	a_1	d_1	l_2
i_2	a_3	d_1	l_2

$\Pi = 0.7$

#i	T-a	Date	Lieu
i_1	a_1	d_1	l_2
i_2	a_3	d_1	l_2

$\Pi = 0.3$

#i	T-a	Date	Lieu
i_1	a_1	d_1	l_2
i_2	a_4	d_1	l_2

$\Pi = 0.3$

#i	T-a	Date	Lieu
i_1	a_1	d_3	l_2
i_2	a_4	d_1	l_2

Π Indique le degré de possibilité du monde considéré

Extension des requêtes algébriques aux données mal connues de types possibilistes

#i	T-a	Date	Lieu
i_1	a_1	$1/d_3 + 0.7/d_1$	l_1
i_2	$1/a_3 + 0.3/a_4$	d_1	l_2

T-a	lg	vt
a_1	20	1000
a_2	25	1200
a_3	18	800
a_4	20	1200

- Ex: dans quelle mesure est-il possible que le n-uplet <d1, 20> appartienne au résultat de la requête Q donnant les paires (d,l) telles qu'il existe (au moins) une image prise à la date d et représentant un avion de longueur l

$\Pi = 1$

#i	T-a	Date	Lieu
i_1	a_1	d_3	l_1
i_2	a_3	d_1	l_1

$\Pi = 0.7$

#i	T-a	Date	Lieu
i_1	a_1	d_1	l_1
i_2	a_3	d_1	l_1

$\Pi = 0.3$

#i	T-a	Date	Lieu
i_1	a_1	d_3	l_1
i_2	a_4	d_1	l_1

$\Pi = 0.3$

#i	T-a	Date	Lieu
i_1	a_1	d_1	l_1
i_2	a_4	d_1	l_1

Extension des requêtes algébriques au cas des données multimédia

- Une requête
  - Décrite par un ensemble de prédicats qui sont combinés dans une formule f selon un syntaxe f ::= p|f ∧ f|f ∨ f | ¬f |(f) où f est une formule et p est un prédicat
- Un tuple t,
  - un attribut flou de ce tuple t est formé par deux composants A^v (valeur) et A^u (probabilité).
  - Respect d'un tuple d'un ensemble de prédicats apparu dans une requête est décrit par une probabilité de satisfaction s(f, t).

EX:

Pid	Title	Author	Color	*u*
P001	Adorazione dei Magi	Leonardo	Red: 0.8	0.58
P002	Battesimo di Cristo	Leonardo	Red: 0.5	0.5

Attribut flou

Tuple flou

	FS	FA
$s(f_1 \wedge f_2, t)$	$Min(s(f_1, t), s(f_2, t))$	$s(f_1, t).s(f_2, t)$
$s(f_1 \vee f_2, t)$	$Max(s(f_1, t), s(f_2, t))$	$s(f_1, t) + s(f_2, t) - s(f_1, t).s(f_2, t)$
$s(\neg f, t)$	$1 - s(f, t)$	$1 - s(f, t)$

Conjonction

Extension de l'algèbre relationnelles aux symboliques

- Un tuple t,
  - $s(f(pi, ..., pn), t) = s_f(s(p1, t), ..., s(pn, t))$
  - $x_i = s(p_1, t)$, $\Theta = [\theta_1, ..., \theta_n]$ avec $\theta i \in [0, 1], \sum_i \theta_i = 1$,
    $s_f(x_1, ..., x_n) = (\theta_1 - \theta_2).x1 + 2(\theta_2 - \theta_3).sf(x1, x2) + L + n.\theta n.sf(x1, ..., xn)$

	FS	FA
$s(f_1 \wedge f_2, t)$	$Min(s(f_1, t), s(f_2, t))$	$s(f_1, t) . s(f_2, t)$
$s(f_1 \vee f_2, t)$	$Max(s(f_1, t), s(f_2, t))$	$s(f_1, t) + s(f_2, t) - s(f_1, t) . s(f_2, t)$
$s(\neg f, t)$	$1 - s(f, t)$	$1 - s(f, t)$