Bases de Données II, Charleroi

Jef Wijsen

12 janvier 2009

Répondez aux huit questions dans les espaces réservés. Durée : 3 heures

Nom et prénom
Année

La Figure 1 montre une base de données XML avec des informations sur des films. Il s'agit d'une liste d'acteurs suivie d'une liste de films:

- Chaque acteur est identifié par un code unique; c'est l'attribut `id`. L'attribut `mort` est seulement présent si l'année de décès est connue.
- Pour chaque film, on enregistre le cast (i.e. les acteurs) en utilisant les codes des acteurs.

La Figure 2 montre le DTD.

Question 1 *(3 points)* Écrivez une requête **XQuery** qui rend tous les films *dans l'ordre de leur année d'apparition*. L'output est formaté comme un document XML, comme suit :

```
<FILMS>
   <film>McVicar</film>
   <film>Chariots of Fire</film>
   <film>Empire of the Sun</film>
</FILMS>
```

```
<?xml version="1.0"?><!DOCTYPE filmotheque SYSTEM "films.dtd">
<filmotheque>
<ACTEURS>
   <acteur id="IC" gendre="M" naissance="1949" mort="1990">Ian Charleson</acteur>
   <acteur id="CC" gendre="F" naissance="1949">Cheryl Campbell</acteur>
   <acteur id="NH" gendre="M" naissance="1949">Nigel Havers</acteur>
   <acteur id="BM" gendre="M" naissance="1950">Bill Murray</acteur>
   <acteur id="MR" gendre="F" naissance="1959">Miranda Richardson</acteur>
   <acteur id="JM" gendre="F" naissance="1960">Julianne Moore</acteur>
</ACTEURS>
<FILMS>
   <film annee="1981">
      <titre >Chariots of Fire</titre>
      <directeur naissance="1936">Hugh Hudson</directeur>
      <cast> <acteur id="IC"/> <acteur id="CC"/> <acteur id="NH"/> </cast>
   </film>
   <film annee="1980">
      <titre>McVicar</titre>
      <directeur naissance="1936">Tom Clegg</directeur>
      <cast> <acteur id="CC"/> <acteur id="BM"/> </cast>
   </film>
   <film annee="1987">
      <titre >Empire of the Sun</titre>
      <directeur naissance="1946">Steven Spielberg</directeur>
         <cast> <acteur id="MR"/> <acteur id="NH"/> </cast>
   </film>
</FILMS>
</filmotheque>
```

Figure 1: Fichier XML avec des informations sur des films.

Question 2 *(3 points)* Écrivez une expression **XPath** qui rend les titres des films dont le régisseur est né en 1936. Il y en a deux :

```
<titre>Chariots of Fire</titre>
<titre>McVicar</titre>
```

```
<!-- This file is called films.dtd -->
<!ELEMENT filmotheque (ACTEURS, FILMS)>
<!ELEMENT ACTEURS (acteur)*>
<!ELEMENT FILMS (film)*>
<!ELEMENT film (titre, directeur, cast)> <!ATTLIST film annee CDATA #REQUIRED>
<!ELEMENT cast (acteur)*>
<!ELEMENT acteur (#PCDATA)>              <!ATTLIST acteur id CDATA #REQUIRED>
                                         <!ATTLIST acteur naissance CDATA #IMPLIED>
                                         <!ATTLIST acteur mort CDATA #IMPLIED>
                                         <!ATTLIST acteur gendre CDATA #IMPLIED>
<!ELEMENT titre (#PCDATA)>
<!ELEMENT directeur (#PCDATA)>           <!ATTLIST directeur naissance CDATA #REQUIRED>
                                         <!ATTLIST directeur mort CDATA #IMPLIED>
```

Figure 2: DTD.

Question 3 *(3 points)* Écrivez une expression **XPath** qui rend les titres des films dans lesquels a joué Nigel Havers. L'expression doit rester valide si on change l'identifiant de Nigel Havers (par exemple, si on remplaçait NH par NiHa). Il y en a deux :

```
<titre>Chariots of Fire</titre>
<titre>Empire of the Sun</titre>
```

Question 4 *(2 points)* Traduisez l'expression XPath suivante en français simple.

```
//film[cast/acteur/@id=/filmotheque/ACTEURS/acteur[@mort]/@id]/titre
```

Question 5 *(7 points)* Écrivez un programme **XSLT** qui rend tous les noms d'acteur et, pour chaque acteur, les titres de tous les films dans lesquels il a joué. L'output est formaté comme un document XML, comme suit :

```
<ACTEURS>
<acteur><nom>Ian Charleson</nom><FILMS><film>Chariots of Fire</film></FILMS></acteur>
<acteur><nom>Cheryl Campbell</nom><FILMS><film>Chariots of Fire</film>
                                        <film>McVicar</film></FILMS></acteur>
<acteur><nom>Nigel Havers</nom><FILMS><film>Chariots of Fire</film>
                                      <film>Empire of the Sun</film></FILMS></acteur>
<acteur><nom>Bill Murray</nom><FILMS><film>McVicar</film></FILMS></acteur>
<acteur><nom>Miranda Richardson</nom><FILMS><film>Empire of the Sun</film></FILMS></acteur>
<acteur><nom>Julianne Moore</nom><FILMS></FILMS></acteur>
</ACTEURS>
```

Question 6 *(6 points)* Le magazine *Top Sport* met en ligne un vaste nombre d'articles sur tous les sports. Un adepte de tennis s'intéresse aux articles sur le tennis, et rien qu'aux articles sur le tennis. Au lieu de chercher ces articles "à la main", il envisage de construire un classificateur pour classer les articles en deux classes : ceux qui traitent du tennis (Tennis="oui") et les autres (Tennis="non"). Pour ce faire, il dispose d'une table qui enregistre le nombre d'occurrences de certains mots clé dans chaque article. Par exemple, l'article `doc1.pdf` contient 12 fois le mot "Saive", 0 fois le mot "Henin", etc.; cet article ne relève pas du tennis.

Article	#Saive	#Henin	#Wimbledon	#Beijing	...	Tennis
doc1.pdf	12	0	0	5		non
doc2.pdf	0	17	1	3		oui
			⋮			⋮

Les matrices de confusion se présentent comme suit:

		Predicted	
		Tennis=oui	Tennis=non
Observed	Tennis=oui	*TP*	*FN*
	Tennis=non	*FP*	*TN*

À partir d'une telle matrice, deux mesures de qualité sont calculées:

$$P = \frac{TP}{TP + FP} \qquad \text{et} \qquad R = \frac{TP}{TP + FN}$$

Sur un ensemble de test, on obtient les valeurs de P et R suivantes pour trois programmes de classification:

	P	R
Naive Bayes	0.80	0.80
C4.5	0.90	0.20
MultilayerPerceptron	0.20	0.90

Si vous deviez choisir un modèle de prédiction à partir de ces chiffres, quel serait ce choix (cocher une case) ?

□ Naive Bayes □ C4.5 □ MultilayerPerceptron

Justifiez votre choix en détail.

Question 7 *(6 points)* Le *diamètre* d'un ensemble C de points est défini comme suit :

$$\mathit{diametre}(C) = \max\{\mathit{distance}(p,q) \mid p,q \in C\} \ ,$$

où $\mathit{distance}(p,q)$ est la distance entre deux points p et q (selon une notion de distance prédéterminée). La *faiblesse* d'un k-clustering $\mathbb{C} = \{C_1, \ldots, C_k\}$ est définie comme

$$\mathit{faiblesse}(\mathbb{C}) = \max_{1 \le i \le k} \mathit{diametre}(C_i) \ .$$

Donc, la faiblesse d'un k-clustering est la distance maximale entre deux points appartenant à un même cluster. L'objectif sera de trouver un clustering de faiblesse minimale. Deux méthodes de clustering hiérarchique sont connues comme *single link* et *complete link*.

1. *(3 points)* Quelle de ces deux méthodes est préférable pour atteindre l'objectif de minimiser la faiblesse. Cochez une case: ☐ single link ☐ complete link

   Expliquez de façon détaillée et précise.

2. *(1 point)* Est-ce que la réponse à la première question pourrait varier selon la notion de distance utilisée ? Cochez une case: ☐ oui ☐ non

   Expliquez.

3. *(2 points)* Dans la Table 8.5, les rangées pour *single link* et *complete link* diffèrent seulement en γ. Expliquez cette différence de façon détaillée et précise.

Question 8 *(10 points)*

TID	items bought
1	{dinde, bière, ail, coca}
2	{bière, endive, ail}
3	{dinde, bière, ail, coca}
4	{dinde, endive, ail, coca}
5	{bière, endive, ail, coca}
6	{bière, ail, coca}
7	{endive, ail}
8	{dinde, bière, endive}
9	{dinde, ail, coca}
10	{bière, ail}

Le support seuil est de 0.25 (c'est-à-dire, 25%).

1. *(4 points)* Montrez les résultats intermédiaires et finaux d'une recherche *depth-first* "intelligente" qui trouve tous les *frequent itemsets*. Une approche "force brute" qui énumère simplement *tous* les itemsets et calcule leur support, n'est pas "intelligente".

2. *(3 points)* Pour la première branche en profondeur (et seulement pour cette branche), dessinez les *FP-trees* utilisés dans l'exécution de FP-growth.

3. *(1 points)* Donnez les *maximal frequent itemsets*.

4. *(2 points)* Donnez les *non-closed frequent itemsets*, i.e. les *frequent itemsets* qui ne sont pas *closed* (il y en a quatre). Expliquez pourquoi ces quatre *frequent itemsets* ne sont pas *closed*.