

Treillis de Galois et Analyse Formelle de concepts

Découverte des treillis avec Conexp ou RCAexplore

Construction d'une ontologie tirée d'un treillis

1 Installation de Conexp

Dans ces travaux, nous utiliserons :

- le logiciel **Conexp** que vous trouverez à l'URL <http://conexp.sourceforge.net/>, pour construire des treillis, des règles d'implication et d'association, et pratiquer l'exploration d'attributs, utilisable avec Java 1.7 pour la partie graphique,
- le logiciel **RCAexplore** que vous trouverez à l'URL <http://dataqual.engees.unistra.fr/logiciels/rcaExplore>, pour construire des treillis, des AOC-posets, ou des treillis Iceberg,
- le logiciel **Protégé**, installé sur les ordinateurs de la Faculté Des Sciences, pour la construction de l'ontologie.
- Open office, Libre office ou Excel : Il est en effet recommandé de saisir vos données brutes dans une feuille de calcul afin de les manipuler avec facilité puis de les transformer dans les formats d'entrée des logiciels cibles.

Vous rédigerez un compte-rendu d'expérience incluant toutes vos données et tous vos résultats (il peut être rendu par groupes de 2 ou 3 étudiants).

2 Cartes de pizza

Dans cette activité, nous vous proposons de construire un treillis dans lequel vous analyserez les cartes de deux pizzerias (ou plus si vous êtes inspirés). Vous utiliserez des cartes de pizza disponibles sur la toile, par exemple :

- <http://annexe-pizza.com/>
- <https://www.pizzahut.fr/>
- <http://www.dominos.fr/>

Les étapes du travail consistent à :

- effectuer un choix de pizzas sur les cartes, en gardant des pizzas blanches et des pizzas rouges (au moins 5 de chaque).
- identifier les attributs pertinents pour décrire les pizzas par leurs ingrédients mais peut-être aussi par d'autres caractéristiques, telles que la forme en chausson (pour les pizzas calzone), les pâtes fines, les pâtes épaisses, les prix, etc.
- identifier une hiérarchie de spécialisation sur les attributs choisis, par exemple **gruyère** et **emmental** sont des spécialisations de **fromage**. Identifiez les ingrédients végétariens, végétaliens, etc.
- construire un (ou plusieurs) treillis : au moins un treillis contenant les pizzas des deux pizzerias (préfixez dans ce cas le nom de la pizza par le nom du pizzeria qui l'a conçue), mais peut-être aussi un treillis par pizzeria.
- analyser vos données.
- construire avec **Protégé** une ontologie des pizzas tirée de vos observations. Lors du prochain cours, nous verrons un procédé systématique, mais ici vous le ferez de manière empirique afin de comprendre les questions posées par une telle construction.

Dans l'analyse des données, nous demandons de répondre aux questions suivantes :

1. Retrouvez-vous certaines des catégories de pizzas utilisées par les pizzerias ? Ex. "Les nouveautés", "Les tentations", "Les grandes faims", "Viandes", "Fromage", "Marée", "Pizzas végétariennes" (à adapter selon ce que vous parcourez comme site) ? Pourquoi ?
2. Parfois il n'y a pas de catégories, utilisez le treillis pour définir des catégories de pizzas qui pourraient servir dans les deux cartes pour présenter les pizzas aux consommateurs. Quels critères utilisez-vous ?

3. Montrer comment, si vous êtes habitués à une pizza chez un pizaiolo, le treillis peut vous aider à trouver une pizza proche chez l'autre pizaiolo. Que veut dire "proche" pour vous ? Quelles sont les différences entre les deux pizzas et où les voyez-vous dans le treillis ?
4. Comment trouver une pizza pour quelqu'un qui est allergique aux fruits de mer ? Qui est végétarien ? Végétalien ?
5. Extrayez quelques règles pertinentes sur le domaine des pizzas (expliquez comment).
6. Montrez sur quelques exemples bien choisis comment cela se modéliserait dans une logique de description.

Annexe : Installation et utilisation de RCAexplore

Vous trouverez l'outil RCAexplore (créé par Xavier Dolques) à l'adresse :

— <http://dolques.free.fr/rcaexplore/> (récupérez et renommez le binaire en `rcaexplore.jar`)

La documentation vous montrera comment lancer l'outil et l'utiliser avec une interface graphique et ci-dessous vous avez les principales commandes pour l'utiliser en ligne de commandes.

Comme il s'agit d'un prototype de recherche, comme pour les autres outils, faites régulièrement des sauvegardes dans différents fichiers pour sécuriser vos données.

2.1 L'éditeur de famille de contextes

On le lance en ligne de commande ainsi :

```
java -jar rcaexplore.jar editor &
```

Il permet de créer des familles de contextes ; il charge et sauve des fichiers dans un format texte lisible et éditable facilement, le format RCFT (extension des fichiers `.rcft`).

2.2 Le générateur de treillis de concepts, AOC-posets, et treillis Iceberg

Vous trouverez à la fin du document des exemples de fichiers d'entrée.

On lance le générateur en ligne de commande par :

```
java -jar rcaexplore.jar explogui <fichier.rcft> <dossier de sortie>
```

pour lancer RCA exploratoire sur le fichier `rcft` avec une interface graphique

En choisissant les options correspondantes au démarrage, l'exploration produit le fichier `result.xml` et divers autres fichiers qui peuvent vous intéresser. Elle produit toujours un fichier `.dot` contenant le treillis, un fichier `trace.csv` qui contient les options de configuration de chaque étape, et un fichier `latticebuilder.sh` qui sert à générer des pdf pour visualiser les treillis.

Observez la configuration courante, qui est celle de l'étape 0, par `display configuration`. Vous pourrez la changer à chaque étape lorsque vous serez en mode manuel, en choisissant au début les contextes objet-attribut, puis ensuite ces mêmes contextes ainsi que les contextes objet-objet (qui apparaissent plus tard).

On peut créer (`choose construction algorithm`) des treillis (FCA) des sous-hiérarchies de Galois (GSH, encore appelés AOC-posets, par l'algorithme ares) et des treillis Iceberg. `icebergXX` correspond au treillis de concepts dont on n'a conservé que les concepts dont la taille de l'extension est supérieure ou égale à XX% du nombre total d'objets. On garde aussi le concept bottom.

`continue` permet d'aller d'une étape à la suivante en changeant la configuration si vous restez en mode `manual`. Sinon, toute la construction (par `auto`) se fait avec la configuration de départ.

À partir de l'étape 1, en mode manuel, vous pouvez choisir les contextes mais aussi les opérateurs de scaling qui vous intéressent.

En cours de route, vous pouvez examiner les treillis construits (dans le répertoire `<dossier de sortie>`).

Les fichiers `stepi.dot` et `stepi-j.dot` vous présentent les structures construites à chaque étape de l'exploration par RCA. Ils peuvent être transformés en fichiers SVG, PDF, PNG, PS ou autres formats grâce à la commande `dot` en ligne de commande (logiciel GraphViz). Le fichier `latticebuilder.sh` contient les commandes pour créer ces fichiers sans écrire vous-même toutes les commandes `dot` nécessaires.

Un exemple de commande est donné ci-dessous si vous voulez le faire vous-même (transformation en fichier pdf) :

```
dot -Tpdf stepi-j.dot -o stepi-j.pdf
```

A la fin, `result.xml` correspond à l'ensemble des structures conceptuelles générées pour toutes les étapes du processus, et une option de visualisation vous montre les structures directement. Il peut être analysé automatique pour tirer différents résultats a posteriori.

Pour résumer, le mode automatique permet de lancer toutes les étapes de RCA. Le mode manuel permet d'ajouter et de retirer à chaque étape des contextes et des relations en surveillant que c'est réalisé de manière cohérente. Il permet aussi de modifier à chaque étape les choix d'opérateurs de scaling et de structure (treillis, treillis Iceberg ou AOC-poset).

2.3 Le navigateur de concepts

Il permet de naviguer de concept à concept. On le lance en ligne de commande uniquement :

```
java -jar rcaexplore.jar browser <fichier.xml>
```

Pour voir le contenu d'un concept il faut choisir l'étape, le contexte et taper le nom du concept puis faire update. Les noms des concepts d'un contexte sélectionné sont affichés avec (entre parenthèses) leur support (taille de l'extension). Il est possible de faire un filtrage dans cette liste pour ne voir que les concepts dont le support est supérieur à une valeur donnée. On peut voir à partir d'un concept ses parents, ses enfants, son extension, son intension simplifiée et son intension. Si on sélectionne un élément de l'intension simplifiée et un élément de l'intension on peut voir juste au-dessous une représentation de la règle d'implication correspondante.