FCP-Growth, une adaptation de FP-Growth pour générer des règles d'association de classe

Emna-Bahri*, Stéphane-Lallich*

*Laboratoire ERIC, Université de Lyon ; 5, avenue Pierre Mendès-France, 69500, Bron
emna.bahri|stephane.lallich@univ-lyon2.fr ; http://eric.univ-lyon2.fr

Motivations. La classification associative (Liu et al., 1998) prédit la classe à partir de règles d'association particulières, dites règles d'association de classe. Ces règles, dont le conséquent doit être la variable indicatrice de l'une des modalités de la classe, s'écrivent $A \rightarrow c_i$, où A est une conjonction de descripteurs booléens et c_i est la variable indicatrice de la i_e modalité de classe. L'intérêt des règles de classe est de permettre la focalisation sur des groupes d'individus, éventuellement très petits, homogènes du point de vue des descripteurs et présentant la même classe. Pour extraire les règles de classe, les méthodes de classification associative procèdent par filtrage des règles générées par les algorithmes d'extraction de règles d'association développés en non-supervisé. Dans une première étape, ces algorithmes extraient tous les itemsets plus fréquents que le seuil, puis ils en déduisent toutes les règles dont la confiance dépasse le seuil de support, ce qui pose différents problèmes. Dans la première étape, on extrait des itemsets fréquents inutiles, ceux qui ne contiennent pas la classe, alors que la seconde étape peut être simplifiée, puisqu'un itemset contenant la classe ne donne lieu qu'à une seule règle de classe. Afin de pouvoir travailler avec des seuils de support le plus bas possible, nous proposons FCP-Growth une adaptation de FP-Growth qui élimine les itemsets fréquents ne contenant pas de classe. En outre, pour ne pas désavantager la classe la moins nombreuse, le seuil de support utilisé dans chaque classe est proportionnel à la taille de la classe.

Etat de l'art. A l'opposé d'Apriori qui génère des itemsets candidats et qui les teste pour ne conserver que les itemsets fréquents, FP-Growth (Han et al. (2000)) construit les itemsets fréquents sans génération de candidats. Tout d'abord, il compresse les itemsets fréquents représentés dans la base de données à l'aide des FP-Tree (*frequent-pattern tree*) dont les branches contiennent les associations possibles des items. Chaque association peut être divisée en fragments qui constituent les itemsets fréquents. La méthode FP-Growth transforme le problème de la recherche de l'itemset fréquent le plus long par la recherche du plus petit et sa concaténation avec le suffixe correspondant (le dernier itemset fréquent de la branche aboutissant à l'item considéré). Ceci permet de réduire le coût de la recherche. Dans notre étude, nous avons retenu FP-Growth, en raison de sa structuration (FP-Tree) qui le rend plus efficace qu'Apriori.

Contribution : FCP-Growth. FCP-Growth, l'algorithme que nous proposons pour construire directement les itemsets de classe fréquents, repose sur plusieurs principes :

- au cours de la construction du FP-Tree, il élimine les itemsets qui ne sont pas de classe, Le gain de temps d'exécution et de stockage obtenu doit permettre de diminuer le seuil de support, ce qui nous aidera à trouver des pépites de classe).
- il utilise un seuil de support adaptatif au sens où le seuil utilisé dans chaque classe est proportionnel à la taille de la classe, dans le but de ne pas pénaliser les petites classes.

FCP-Growth

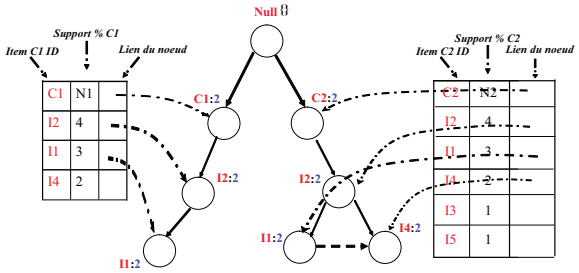

FIG. 1 – *Construction de FCP-tree*

Résultats. Pour évaluer l'efficacité de FCP-Growth, nous l'avons testé sur 3 bases de données réelles volumineuses.Sur l'ensemble des trois bases traitées, il apparaît qu'entre le tiers et la moitié des itemsets générés par FP-Growth ne sont pas pertinents, ce qui alourdit inutilement la procédure. Grâce à sa procédure d'élimination des itemsets qui ne sont pas de classe et à son seuil adaptatif, FCP-Growth permet d'extraire plus de règles de classes que FP-Growth : le taux sur les trois bases de données varie de 13% à 52%, suivant le seuil, tout en évitant que la classe minoritaire soit trop défavorisée. Le taux de couverture, qui indique la proportion d'exemples qui sont couverts par au moins un itemset de classe, est nettement augmenté par FCP-Growth pour arriver à environ 90%, ce qui représente entre 19% et 35% d'augmentation de selon les bases. Alors que FCP-Growth permet de traiter globalement moins d'itemsets que FP-Growth, le temps d'exécution de FCP-Growth est toujours inférieur ou égal à celui de FP-Growth, d'autant plus que le seuil de support est petit. En outre, cette comparaison ne prend en compte que le temps d'exécution nécessaire à la construction des itemsets et néglige le temps nécessaire au filtrage des itemsets de classe lorsque l'on utilise FP-Growth.

Conclusion et perspectives. Ce travail propose FCP-Growth, une adaptation de FP-Growth à la recherche des itemsets et règles d'association de classe. FCP-Growth construit les seuls itemsets fréquents de classe, en se basant sur un support qui s'adapte à la taille de chaque classe pour éviter de défavoriser les classes minoritaires. Les résultats trouvés montrent un e amélioration du taux de couverture ainsi qu'un gain de temps et de stockage grâce à la génération des seuls itemsets fréquents de classe, ce qui permettra de diminuer le seuil de support. Ces résultats justifient l'intégration de FCP-Growth comme algorithme de génération de règles dans une procédure de classification associative.

Références

Han, J., J. Pei, et Y. Yin (2000). Mining frequent patterns without candidate generation. In W. Chen, J. Naughton, et P. A. Bernstein (Eds.), *2000 ACM SIGMOD Intl. Conference on Management of Data*, pp. 1–12. ACM Press.

Liu, B., W. Hsu, et Y. Ma (1998). Integrating classification and association rule mining. In *Knowledge Discovery and Data Mining*, pp. 80–86.