

Elicitation de préférences

Direction: Sofian Maabout

Dans les bases de données, le résultat d'une requête peut être trop volumineux pour être facilement exploitable, voire utile. Ainsi, il devient important de filtrer ce résultat d'une manière ou d'une autre afin de ne retenir que la partie la « plus intéressante ». Un exemple courant de ce type de procédé est celui que l'on rencontre quand on utilise un moteur de recherche. Très souvent, les résultats les plus intéressants se trouvent *en tête de liste*. Cela signifie simplement que ce moteur de recherche a *trié* les résultats selon certains critères. On peut alors parler de score. Plusieurs travaux ont porté sur l'évaluation des requêtes retournant les *k-meilleurs* parmi *n* résultats possibles. On parle alors de requêtes *Top-k* [1, 2]. Ces dernières nécessitent généralement l'introduction d'une fonction de score qui consiste à combiner les valeurs prises par les attributs décrivant un enregistrement et retourne une valeur numérique qui permettra de sélectionner justement les *k* enregistrements qui ont le meilleur score.

Dans certaines situations, il est difficile, voire impossible, de définir une fonction de score. Par exemple, choisir la meilleure chambre d'hôtel quand les chambres sont décrites par le prix et l'éloignement par rapport à un arrêt de bus. Dans ces cas, on cherche les chambres pour lesquelles il n'existe aucune autre chambre qui leur soit meilleure du point de vue prix ET du point de vue proximité d'un arrêt de bus. On parle alors de relation de dominance. La requête qui consiste à retourner les objets non dominés s'appelle *Skyline* [3], ou front de Pareto. Dans ce cas, l'utilisateur n'a plus besoin de combiner les différents attributs pour définir un score par contre on a besoin de définir un ordre (possiblement partiel) sur les valeurs de chaque attribut. Ces différents ordres expriment des *préférences* [4].

Dans le cadre de cette thèse, on se place dans un contexte où l'utilisateur n'a pas besoin de spécifier ces ordres. On donnera plutôt des instances de la relation de dominance à partir desquelles on va *inférer (éliciter)* ces ordres. Ces derniers seront alors explicités à l'utilisateur qui pourra par la suite y apporter. Il s'agit là de la première partie du travail envisagé dans le cadre de cette thèse.

Une deuxième partie pourra porter sur *l'agrégation de skylines*. Par exemple, dans un réseau social, les personnes faisant partie d'une même communauté peuvent avoir des préférences musicales différentes. On pourrait par exemple avoir autant d'ordres différents que de personnes dans la communauté. Pour chaque personne, on peut donc définir un skyline. La question que l'on peut se poser est celle de trouver le ou les skylines, donc l'ensemble des ordres sur chacun des attributs, qui représente le meilleur consensus.

Bibliographie :

[1] Ronald Fagin: Combining Fuzzy Information from Multiple Systems. J. Comput. Syst. Sci. 58(1): 83-99 (1999)

[2] Reza Akbarina, Esther Pacitti, Patrick Valduriez: Best position algorithms for efficient top-k query processing. Inf. Syst. 36(6): 973-989 (2011)

[3] Jongwuk Lee, Seung-won Hwang: BSkyTree: scalable skyline computation using a balanced pivot selection. EDBT 2010: 195-206

[4] W. Kießling. Foundations of Preferences in Database Systems. In International Conference on Very Large Data Bases (VLDB), pages 311–322, 2002.