

Wave Top-k Random-d Family Search: comment guider un expert dans un espace structuré (matériel supplémentaire)

Résumé. Dans cet article, nous développons une méthode (WTRFS) incluant le retour utilisateur permettant de le guider parmi les résultats d'une fouille de motifs. Ce travail vise à remplacer l'étape de déclaration des descripteurs utilisée dans la fouille interactive de motifs, en s'appuyant sur l'existence hypothétique d'un lien entre les différents motifs intéressants un expert. Nous montrons empiriquement que WTRFSretourne rapidement les résultats les plus pertinents pour l'utilisateur. De plus, même si les retours d'utilisateur ne sont pas parfaits, le comportement de WTRFSn'est pas altéré.

Annexes

A Algorithme de pondération

Dans l'algorithme 1, on commence par initialiser l'ensemble de sommets portant les modifications \mathbb{V}' avec l'ensemble des sommet \mathbb{V} , l'ensemble des ancêtres du sommet v noté L_a avec l'ensemble des parents de v noté $\mathcal{P}(\{v\})$, et l'ensemble des descendants du sommet v noté L_d avec l'ensemble des enfants de v noté $\mathcal{C}(\{v\})$ (ligne 1). On récupère la valeur absolue du poids de v (ligne 2) puis on met à jour le poids de v (ligne 3) (qui peut soit doubler dans le négatif, soit devenir nul, soit doubler dans le positif). Ligne 5 à 15, on répète une boucle tant que l'ensemble des ancêtres et l'ensemble des descendants de v n'a pas été traité. Ligne 6 à 8 et 9 à 11, on itère sur tous les ancêtres et descendants pour modifier leur poids. L'impact du poids de v est diminué de $\frac{1}{2^k}$ en fonction de la distance à v , avec k initialisé à 2 (ligne 2). Cette valeur a été choisie empiriquement. Nous avons aussi expérimenté une variante où le poids diffusés était diminué en fonction du nombre d'enfants/de parents d'un vertex sans remarquer des différences pratiques. Ligne 13 on récupère les parents des ancêtres contenus dans L_a pour mettre à jour l'ensemble et les enfants des descendants dans L_d pour faire de même. Puis on augmente k de 1 à la ligne 14 afin de réduire l'effet de changement de pondération sur des couches de profondeur supplémentaire.

Algorithm 1 Pondération de la lignée

Require: \mathbb{V} un ensemble de sommets, $v \in \mathbb{V}$ un sommet, et A une interaction expert·e.

Ensure: l'ensemble des sommets où les ancêtres et les descendants de v ont un poids modifié en fonction de A .

```

1:  $\mathbb{V}' \leftarrow \mathbb{V}; L_a \leftarrow \mathcal{P}(\{v\}); L_d \leftarrow \mathcal{C}(\{v\})$ 
2:  $w \leftarrow |\text{poids}(v)|$ 
3:  $\text{poids}(v) = \text{pondération}(v, A, w)$ 
4:  $k \leftarrow 2$ 
5: while  $L_a \neq \emptyset$  or  $L_d \neq \emptyset$  do
6:   for  $a \in L_a$  do
7:      $\text{poids}(a) = \text{pondération}(a, A, w * \frac{1}{2^k})$ 
8:   end for
9:   for  $d \in L_d$  do
10:     $\text{poids}(d) = \text{pondération}(d, A, w * \frac{1}{2^k})$ 
11:   end for
12:    $L_a \leftarrow \mathcal{P}(L_a); L_d \leftarrow \mathcal{C}(L_d)$ 
13:    $k \leftarrow k + 1$ 
14: end while

```

B Algorithme d'échantillonnage

Dans l'algorithme 2 on commence par initialiser les ensembles L^+ et $L^?$ contenant respectivement les éléments prioritaire et sans avis vis à vis de l'exploration ligne 1 et 2. On initialise l'ensemble \mathbb{S} échantillonné à la ligne 3. On effectue les k Top tirage dans les ligne 4 à 9. On effectue les tirages pseudo-aléatoires de la ligne 10 à la ligne 15. On effectue prioritairement les tirages dans les éléments prioritaire dans les conditions lignes 4 et 10. On renvoie l'ensemble des échantillons ligne 16.

Algorithm 2 Échantillonage_{k,d}(L, G)

Require: G le graphe étudié, L une couche du graphe, k le nombre de tirage "top", d le nombre de tirage pseudo-aléatoire.

Ensure: S l'ensemble des sommets échantillonné dans L.

```
1:  $L^+ \leftarrow \{\forall v \in L | v \in \mathbb{V}^+\}$ 
2:  $L^? \leftarrow \{\forall v \in L | v \notin \mathbb{V}^+ \text{ & } v \notin \mathbb{V}^-\}$ 
3:  $S \leftarrow \emptyset, S' \leftarrow \emptyset$ 
4: if  $|L^+| \geq k$  then
5:    $S \cup \{v_1, \dots, v_k \in L^+ | \exists v_{k+1} : f_p(v_{k+1}, G) > f_p(v_i, G), 1 \leq i \leq k\}$ 
6: else
7:    $S \cup \{v_1, \dots, v_{k-|L^+|} \in L^+ | \exists v_j : f_p(v_j, G) > f_p(v_i, G), 1 \leq i \leq k - |L^+|\}$ 
8:    $S \cup \{v_1, \dots, v_{k-|S|} \in L^? | \exists v_j : f_p(v_j, G) > f_p(v_i, G), 1 \leq i \leq k - |S|\}$ 
9: end if
10: if  $|L^+| \geq d$  then
11:    $S' \cup \{v_1, \dots, v_d \in L^+ \text{ aléatoirement choisi selon Équation (4)}\}$ 
12: else
13:    $S' \cup \{v_1, \dots, v_{d-|L^+|} \in L^+ \text{ aléatoirement choisi selon Équation (4)}\}$ 
14:    $S' \cup \{v_1, \dots, v_{d-|S'|} \in L^? \text{ aléatoirement choisi selon Équation (4)}\}$ 
15: end if
16: return  $S \cup S'$ 
```

C Courbes de rappel pour la sélection aléatoire

Dans la figure 1, pour chaque illustration, l'axe des abscisses indique le nombre de motifs proposés à l'oracle, et l'axe des ordonnées indique le pourcentage d'étiquettes découvertes par type d'étiquette. Les colonnes correspondent aux couches du SIPOG et les lignes correspondent aux types d'oracle. Les couleurs correspondent aux types des étiquettes.

On remarque que la courbe des étiquettes *Rejeté* est toujours au dessus des autres lors de l'échantillonnage aléatoire. Cette courbe est suivie soit par celle des éléments *Accepté* soit par celle des éléments *Inintéressant*. Enfin on voit que, hormis pour espaces les plus restreints, aucune courbe n'atteint les 100% d'étiquettes découvertes.

Dans l'ensemble, les résultats de Wave Top-k Random-d Family Search sont meilleurs que les résultats obtenus par un parcours en vague avec échantillonnage aléatoire.

D Résultats obtenus sur BCR-ABL.

Dans cette section, nous appliquons notre méthode à un jeu de données chimiques étudiés au CERMN¹. Le jeu de données étudié est BCR-ABL obtenu de CHEMBL23², est un ensemble de graphes chimiques contenant 1 485 molécules. L'ensemble des sous-graphes motifs extrait est composé de 112 363 sous-graphes étiquetés fréquents. L'extraction des sous-graphes

1. <http://cermn.unicaen.fr/>

2. <https://chembl.gitbook.io/chembl-interface-documentation/downloads>

Wave Top-k Random-d Family Search: matériel supplémentaire

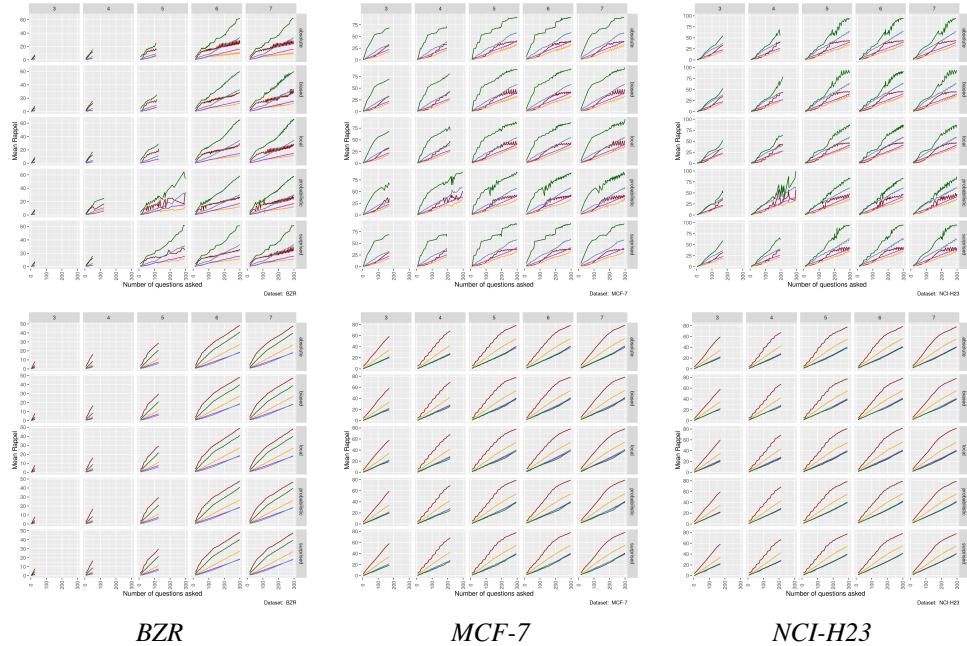


FIG. 1 – Moyenne des pourcentages du Rappel pour BZR, MCF-7, et NCI-H23 avec WTRFSen haut et un parcours en vague avec échantillonage aléatoire en bas.

dont l’ordre est compris entre 1 et 7 a été effectuée avec une fréquence de 10 occurrences par *Norns*³ (Métivier et al., 2018).

Les sous-graphes fréquents, appelés *pharmacophores*, sont des graphes complets dont les sommets sont des marqueurs pharmacophoriques. Les marqueurs pharmacophoriques représentent des propriétés chimiques des molécules influant sur leurs comportements biologique. Chaque pharmacophore voit son support composé de molécules pouvant classifiées comme active ou inactive. Les pharmacophores sont rassemblés dans 1 533 classes d’équivalence identifiées à partir de support identique et d’un lien structurel dans le SIPOG.

Dans ce jeu de données, notre première classe sera donc formée de molécules actives et la seconde classe de molécules inactives. L’activité étant déterminée par rapport à un récepteur.

Dans la figure 2, pour chaque illustration, l’axe des abscisses indique le nombre de motifs proposés à l’oracle, et l’axe des ordonnées indique le pourcentage d’étiquettes découvertes par type d’étiquette. Les colonnes correspondent au couches du POG et les lignes correspondent aux types d’oracle. Les couleurs correspondent aux types des étiquettes indiqués dans le tableau 1 de l’article.

On note, sur cette ensemble de graphe, une forte différence entre les résultats de WTRFS(*a*) et ceux de l’échantillonnage aléatoire (*b*). Lors de l’échantillonnage aléatoire, les étiquettes les plus présentent (*Inintéressant*, *Incertain*, *Intéressant*) sont celles qui sont le plus proposées aux oracles ce qui donne une proportion très basse d’étiquettes *Accepté* et *Rejeté* découvertes. Le

3. <https://valorisation.greyc.fr/catalog/logiciel?identifier=norns>

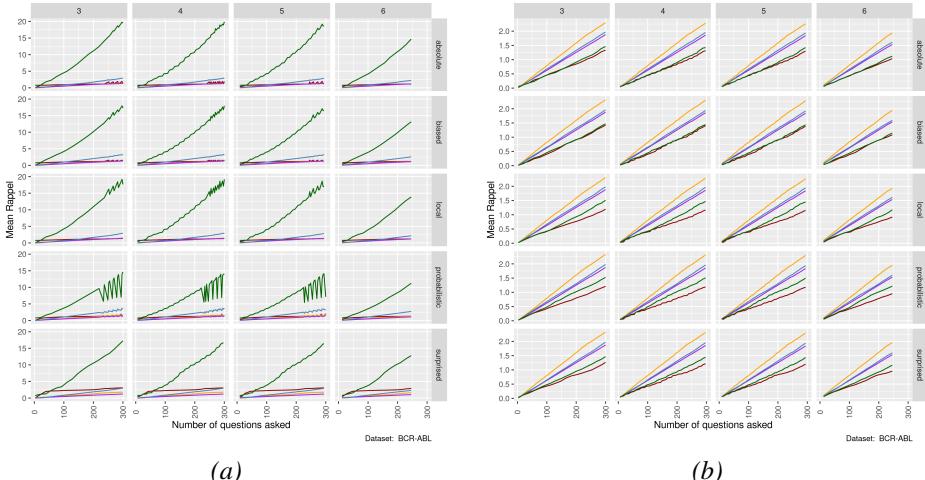


FIG. 2 – Rappel moyen des étiquettes découvertes dans BCR-ABL avec WTRFS(a) et échantillon aléatoire avec parcours en vague (b).

rappel des étiquettes *Rejeté* et *Accepté* a donc du mal à dépasser 1% dans (b) alors qu'il atteint les 20% dans (a).

Références

Métivier, J.-P., B. Cuissart, R. Bureau, et A. Lepailleur (2018). The pharmacophore network : a computational method for exploring structure–activity relationships from a large chemical data set. *Journal of Medicinal Chemistry* 61(8), 3551–3564.

Summary

Assuming there is a structural relation between patterns of interest to an expert, for us the sub-graph relation, we develop a learning method based on the user's feedback guiding them in the results of pattern mining. This strategy aims to replace the standard descriptor declaration step in interactive data mining, which is a source of bias, errors, and lacks flexibility to explore our data once the search has begun. For this, we exploit the structural relationship between patterns modeled by a partial order that is used in the form of a oriented graph structure linking patterns, a partial ordered graph, in order to convert labeled patterns into dynamic descriptors of the solution space and thus of the solution pattern language. Thus the method guides the progressive and interactive definition of pattern descriptors encoding an expert subjective interestingness following the progressive understanding an expert has of the solution pattern space.