YoGa Tool : Base de données et interface web dédiées aux interactions entre l'ontologie et les sites actifs des protéines

Propulsé par

Abstract

YoGa Tool est la conjugaison d'une base de données et d'une interface web. Son but est de permettre l'analyse des interactions entre les ontologies provenant de GO et les sites actifs reconnus par Prosite. Afin d'atteindre ce but, nous avons dû utiliser 3 bases de données : Ensembl, GO and Prosite. Notre interface web autorise plusieurs types de recherches à partir des Access Code mais aussi la création de requêtes complexes spécifiques. Nous avons montré que cette nouvelle approche permet de découvrir de nouvelles connections mais aussi de confirmer de précédentes découvertes. L'outil étant encore à un stade peu avancé, de futurs développements pourront permettre de réaliser des recherches sur un spectre plus étendu, en particulier des recherches sur plusieurs espèces. Un système de mise à jour automatique est aussi à l'étude.

GORET Gaël et L'HOSTIS-JACQUEMIN Yoan
Option B2G – M2_12 MERCB

Master Santé et Populations

Lyon 1

I) Introduction

Les protéines jouent un rôle primordial dans le fonctionnement des organismes, elles sont des acteurs indispensables pour la grande majorité des fonctions cellulaires : les enzymes servent de catalyseurs pour les réactions métaboliques, certaines protéines comme la tubuline ou le collagène assurent la structure des cellules, d'autres encore, comme les cadhérines, jouent un rôle dans la formation des contacts inter-membranaires …

Cette grande variété de fonctions est assurée par une non moins grande variété d'architectures protéique. La structure d'une protéine est en effet adaptée à la fonction qu'elle assure, ce qui a pendant longtemps amené les biologistes à porter leur attention sur l'analyse de motifs structuraux.

L'augmentation de la quantité d'informations présente dans les bases de données de transcrits (Ensembl [1]) et de domaines (Prosite [2]), associée à un outil aussi puissant que Gene Ontology [3] nous permet d'envisager de découvrir de nouvelles relations entre les fonctions des protéines et les sites actifs de protéines indispensables à la réalisation de ces fonctions. De même, il est possible de réaliser la démarche inverse et donc de découvrir dans quelles fonctions globales, un site donné est nécessaire.

Bien que toutes les données nécessaires soient à la disposition des chercheurs, peu d'études ont été réalisées au-delà du problème de l'annotation automatique [4], [5]. Nous avons construit une base de données de tous les transcripts annotés manuellement en association avec leurs fonctions et leurs sites actifs. L'analyse de cette base de données devrait permettre de découvrir de nouvelles interactions et de cibler précisément quels domaines sont nécessaires à une fonction donnée et vice-versa.

II) Matériels et Méthodes

1) Récupération et formatage des données provenant de sites extérieurs

a. Access code et informations basiques via Biomart

Nous avions décidé, dans un premier temps, de nous concentrer sur des transcrits humains, y compris ceux alternativement épissés, codant pour des protéines mais en ne gardant que ceux avec des annotations manuelles. Biomart [6] a permis de réaliser ces extractions de données sur le jeu de données ensembl48 / NCBI36 (Homo sapiens).
La première requête nous a permis de récupérer tous les access code des transcrits ainsi que l'*access code* de leur gène associé (respectivement TAC et GAC). Les deux filtres utilisés ont permis de ne récupérer que les *protein_coding_transcripts* et de ne garder que ceux dont les *molecular_function* possédaient un *evidence_code* différent de IEA (Inferred Automatic Annotation, voir Annexes).
La seconde requête permit de récupérer l'association de ces TAC spécifiques et d'une partie des informations de GO : l'*access code* de GO (GOAC), la description associée (GOD) et l'*evidence code* de cette annotation GOEC).
Et la dernière requête nous a permis de récupérer l'association de ces TAC spécifiques à leurs *access code* Prosite (PAC).

b. Gene Ontology

Etant donné que Biomart ne permet pas d'accéder au type d'ontologie (*molecular function, biological process* et *cellular component*), nous avons dû récupérer la version SQL de GO disponible sur leur site afin de l'intégrer à notre propre base.

c. Prosite

De même Biomart ne permet de récupérer que l'access code de Prosite et non toutes les informations associées. Par conséquent, nous avons récupéré les descriptions des domaines sur le site de Prosite. Cependant Prosite ne propose qu'une version texte de leur base, ce qui nous a obligé à coder un parser en python afin d'intégrer dans notre base les descriptions des domaines et leur motif. Voir Annexes pour plus de détails. Au vu de la taille du fichier source, nous avons fait attention à optimiser le code afin de réduire le temps de traitement.

2) Construction de Schéma relationnel

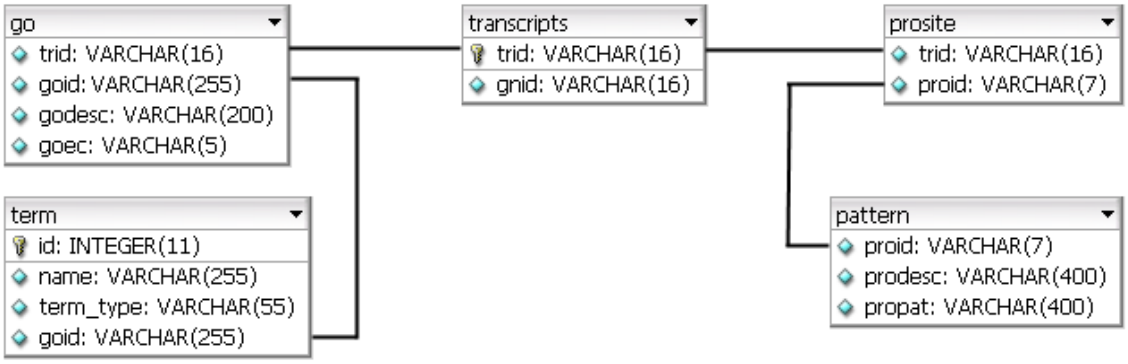


Figure 1. Schéma de la Base de Données

Le schéma de la base de données a été créé afin de faciliter la mise à jour des informations en fonction de chacune des sources. Par exemple, au lieu d'intégrer les descriptions et les motifs des domaines directement dans la table prosite, nous avons choisi de les laisser dans la table pattern (créée à partir du fichier csv de sortie du parser) afin de faciliter sa mise à jour. De même une table a été créée pour chacune des requêtes présentées en II-1 (go, transcripts et prosite) en utilisant l'*access code* des transcripts en tant que clé primaire mais aussi étrangère en l'occurrence.

3) Base de données MySQL et Interface Web

La base de données a été créée en utilisant phpmyAdmin 2.10 et les divers fichiers de données présentés en II-1. L'interface web a été programmée en PHP4 via Notepad++ afin de produire du code HTML et d'accéder à la base de données mySQL.

III) Résultats

1) Interface Web

On peut diviser notre interface web en deux parties : la page d'accueil qui permet de réaliser les recherches et les pages d'affichage des résultats. Il est possible de rechercher dans notre base de données en fonction d'un TAC, d'un GAC, d'un PAC ou d'un GOAC. On peut aussi effectuer des requêtes plus complexes si on connait le langage SQL. Dans ce but, nous avons fourni en tant qu'aide quelques exemples de requêtes et le design de notre base de données. Notre rapport est lui aussi accessible en français et en anglais via cette page d'aide.

Figure 2. Page d'accueil de YoGa Tool permettant de réaliser des recherches sur les TAC (A),GAC (B),PAC (C) et GOAC (D). Possibilité de réaliser aussi des requêtes complexes (E) et d'accéder à l'aide.

Les pages de résultats des requêtes classiques affichent les informations disponibles dans notre base de données accompagnées de liens internes vers notre base de données et de liens externes vers nos bases sources. Un code couleur simple permet de déterminer le type d'ontologie : bleu pour les *Molecular Function*, violet pour les *Cellular Component* et vert pour les *Biological Process.* Lors d'une recherche sur les PAC et GOAC l'*access code* recherché est mis en avant en rouge.

Etant donné qu'il est impossible de prévoir les requêtes complexes, la page de résultat a été codée de manière dynamique. Ce qui signifie qu'elle peut s'adapter quelle que soit la requête demandée. De plus, si l'utilisateur ne change pas le nom des champs dans sa requête le résultat est amélioré par des liens internes et externes. Si on utilise ce type de requête on peut aussi exporter les résultats au format texte, les valeurs séparées par des tabulations. Ceci permet donc de réutiliser ces données dans un logiciel plus avancé tel R.

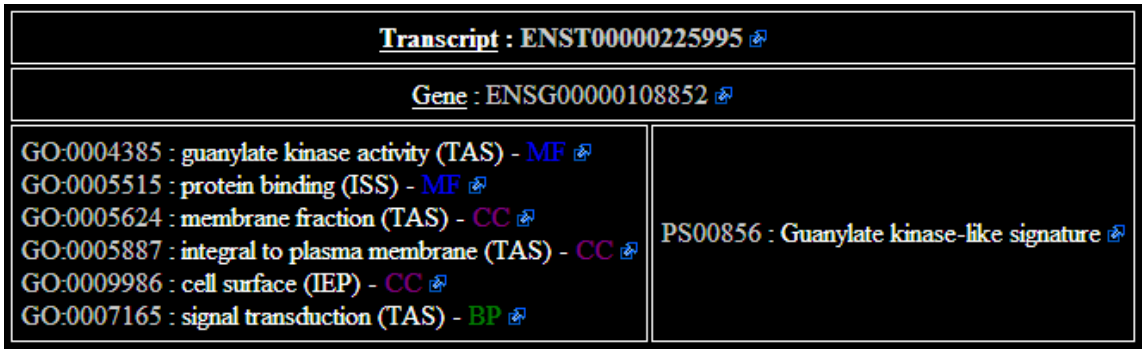


Figure 3. Exemple de résultat de recherche sur un TAC. Les résultats affichent le TAC et les informations GO et Prosite associées.

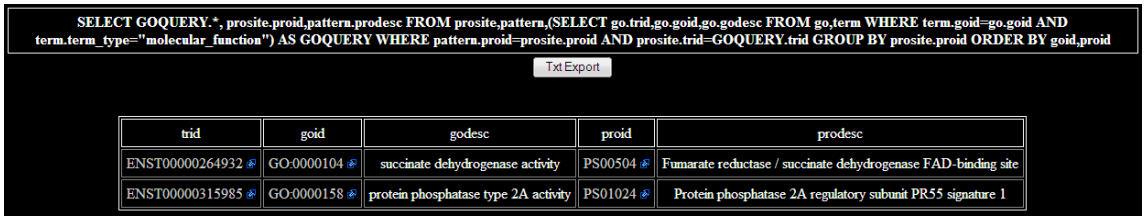
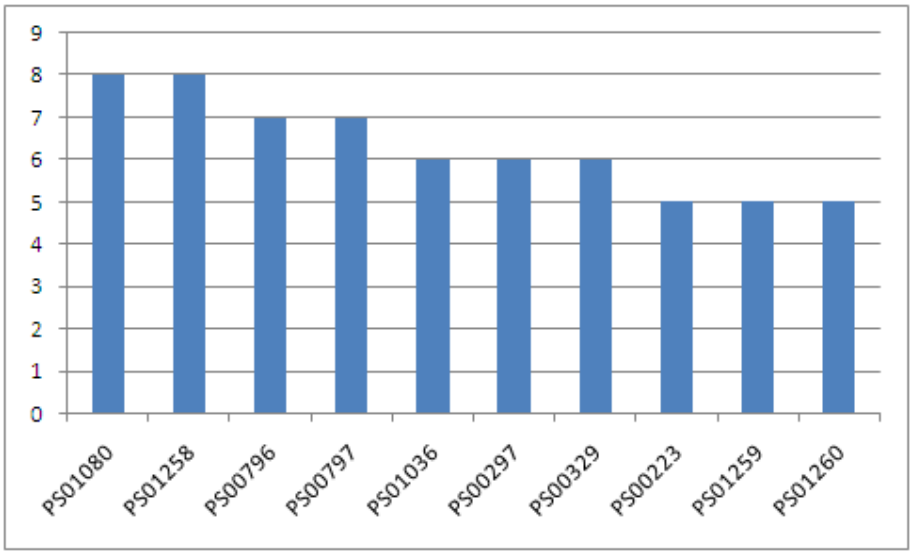


Figure 4. La page de résultat des requêtes complexes affiche dynamiquement les résultats, la requête effectuée et permet l'export au format texte.

2) Exemples d'utilisation de l'export texte

La requête utilisée pour obtenir ces résultats fut la suivante :

```
SELECT COUNT(prosite.proid),prosite.proid,pattern.prodesc
FROM (SELECT * FROM go WHERE goid="GO:0006916")AS ONE, prosite, pattern
WHERE ONE.trid=prosite.trid AND prosite.proid=pattern.proid
GROUP BY prosite.proid ORDER BY COUNT(prosite.proid) DESC
```



Graphique 1. Fréquences des domaines Prosite associés à l'ontologie GO de l'anti-apoptose (GO:0006916) dans la base de données YoGa.

Le graphique 1 montre le nombre d'occurrences des domaines associés à un GOAC spécifique. Ce graphique a été créé grâce à la fonction d'export texte et Excel.

Fréquences	PAC	Description
8	PS01080	Apoptosis regulator, Bcl-2 family BH1 motif signature
8	PS01258	Apoptosis regulator, Bcl-2 family BH2 motif signature
7	PS00796	14-3-3 proteins signature 1
7	PS00797	14-3-3 proteins signature 2
6	PS01036	Heat shock hsp70 proteins family signature 3
6	PS00297	Heat shock hsp70 proteins family signature 1
6	PS00329	Heat shock hsp70 proteins family signature 2
5	PS00223	Annexins repeated domain signature
5	PS01259	Apoptosis regulator, Bcl-2 family BH3 motif signature
5	PS01260	Apoptosis regulator, Bcl-2 family BH4 motif signature

Tableau 1. Fréquences et descriptions des PAC du Graphique 1.

Le Tableau 1 met en relation les TAC observables dans la figure 1 avec leurs fréquences et leurs descriptions.

IV) Discussion

Grâce à notre base de données et à notre interface, nous permettons une nouvelle approche des interactions entre les fonctions des protéines et leurs domaines actifs. Par conséquent on peut espérer découvrir de nouvelles connections insoupçonnées entre ces caractéristiques. De plus, l'accès à des requêtes complexes permet de répondre facilement à des questions difficiles comme, par exemple, de savoir quels domaines protéiques sont impliqués dans certaines fonctions. En examinant les informations données par le Graphique 1 et le Tableau 1 on peut suspecter une interaction entre le processus d'anti-apoptose et la signature des protéines 14-3-3. En effet, de par sa place dans le classement entre 2 domaines déjà connus pour leur rôle dans ce processus (la famille des Bcl-2 et celle des Heatshock hsp70), il serait logique de s'intéresser à l'effet de cette signature dans les interactions avec l'anti-apoptose. De plus, nous avons aussi retrouvé des informations fondamentales telles que l'importance de la liaison avec l'ATP (196 occurrences pour GO et 218 pour Prosite) et des liaisons avec les acides nucléiques (402 occurrences pour GO et 328 pour Prosite) (données non présentées ici).

V) Perspectives

Bien qu'ayant choisi de réduire le champ d'application de notre outil, il est tout à fait envisageable de l'étendre à de plus larges analyses. On peut penser à l'examen d'interactions avec les autres types d'ontologie, ou ne plus se restreindre aux annotations manuelles ou même d'effectuer des analyses multi-espèces.

De plus l'interface web que nous avons codée peut facilement évoluer. En particulier nous avons pensé à la mise en place d'une mise à jour automatique en fonction de celles de nos bases de données source. Cependant la principale difficulté sera d'intégrer à cette mise à jour l'utilisation du fichier plat de Prosite qui nécessite un parsage. Il serait aussi intéressant

d'autoriser des requêtes plus complexes de façon intuitive, comme par exemple de n'afficher que les éléments communs si une liste d'*access code* de transcripts était demandée lors d'une TAC search. Pour terminer nous pensons qu'il serait utile de développer un système de représentation graphique dynamique qui permettrait de mieux visualiser l'ensemble des résultats.

Bibliographie

[1] **T. J. P. Hubbard, B. L. Aken, K. Beal1, B. Ballester1, M. Caccamo, Y. Chen, L. Clarke, G. Coates, F. Cunningham, T. Cutts, T. Down, S. C. Dyer, S. Fitzgerald, J. Fernandez-Banet, S. Graf, S. Haider, M. Hammond, J. Herrero, R. Holland, K. Howe, K. Howe, N.** Ensembl 2007. *Nucleic Acids Res.* 2007, Vol. 35.

[2] **Hulo N., Bairoch A., Bulliard V., Cerutti L., Cuche B., De Castro E., Lachaize C., Langendijk-Genevaux P.S., Sigrist C.J.A.** The 20 years of PROSITE. *Nucleic Acids Res.* 2007 Nov 14.

[3] **Durinck S, Moreau Y, Kasprzyk A, Davis S, De Moor B, Brazma A, Huber W.** BioMart and Bioconductor: a powerful link between biological databases and microarray data analysis. *Bioinformatics.* 2005, Vol. 21, 16.

[4] **Consortium, The Gene Ontology.** Gene Ontology: tool for the unification of biology. *Nature Genet.* 2000, Vol. 25.

[5] http://python.org/. *Python Home.* [Online]

[6] **term, Automatic annotation of protein motif function with Gene Ontology.** Lu X, Zhai C, Gopalakrishnan V, Buchanan BG. *BMC Bioinformatics.* 2004, Vol. 5, 122.

[7] **Schug J, Diskin S, Mazzarelli J, Brunk BP, Stoeckert CJ Jr.** Predicting gene ontology functions from ProDom and CDD protein domains. *Genome Res.* 2002, Vol. 4, 12.

Annexes

1) Format du fichier de donnée Prosite

```
CC   ************************
CC   *** PROSITE data file ***
CC   ************************
CC
CC   Release 20.24 of 04-Dec-2007.
//
ID   ASN_GLYCOSYLATION; PATTERN.
AC   PS00001;
DT   APR-1990 (CREATED); APR-1990 (DATA UPDATE); APR-1990 (INFO UPDATE).
DE   N-glycosylation site.
PA   N-{P}-[ST]-{P}.
CC   /TAXO-RANGE=??E?V;
CC   /SITE=1,carbohydrate;
CC   /SKIP-FLAG=TRUE;
CC   /VERSION=1;
PR   PRU00498;
DO   PDOC00001;
//
ID   CAMP_PHOSPHO_SITE; PATTERN.
AC   PS00004;
DT   APR-1990 (CREATED); APR-1990 (DATA UPDATE); APR-1990 (INFO UPDATE).
DE   cAMP- and cGMP-dependent protein kinase phosphorylation site.
PA   [RK](2)-x-[ST].
CC   /TAXO-RANGE=??E?V;
CC   /SITE=3,phosphorylation;
CC   /SKIP-FLAG=TRUE;
CC   /VERSION=1;
DO   PDOC00004;
//
```

2) Fichier parsé et reformaté à partir de Prosite (csv)

PS00001	N-glycosylation site	N-{P}-[ST]-{P}
PS00004	cAMP- and cGMP-dependent protein kinase phosphorylation site	[RK](2)-x-[ST]
PS00005	Protein kinase C phosphorylation site	[ST]-x-[RK]
PS00006	Casein kinase II phosphorylation site	[ST]-x(2)-[DE]
PS00007	Tyrosine kinase phosphorylation site	[RK]-x(2,3)-[DE]-x(2,3)-Y
PS00010	Aspartic acid and asparagine hydroxylation site	C-x-[DN]-x(4)-[FY]-x-C-…
PS00012	Phosphopantetheine attachment site	[DEQGSTALMKRH]-[L …
PS50075	Acyl carrier protein phosphopantetheine domain profile	NA
PS51257	Prokaryotic membrane lipoprotein lipid attachment site profile	NA
PS51318	Twin arginine translocation (Tat) signal profile	NA
PS00409	Prokaryotic N-terminal methylation site	[KRHEQSTAG]-G …

3) Script de parsing en Python

```
def create_prosite_cvs(nf):

        try:                    # Pour gerer un fichier inconnu
                f = open(nf,'r')
        except IOError, e:      #  gestion des exceptions
                print "Fichier Input inconnu: ", nf
                return

        try:
```

```
        outf = open("prosite_DB.cvs","w")
except IOError, e:       #  gestion des exceptions
        print "Erreur de creation du fichier Output"
        return

row=""
beg=1
l=f.readline()

while  l:
        l=f.readline()

        if l[0:2] not in ["AC","DE","PA"]: #Accelere le Parsing
                continue

        elif l[0:2]=="AC":
                row = row +'"'+ l[5:len(l)-2]+ '";'
                continue

        elif l[0:2]=="DE":
                row = row + '"' +l[5:len(l)-2] +  '";'
                l=f.readline()

                if l[0:2]!="PA":
                        row = row + '"NA"\n'
                        outf.write(row)
                        row=""
                        continue

                else :
                        row = row + '"' + l[5:len(l)-2]
                        l=f.readline()
                        while l[0:2]=="PA" :
                                row = row + l[5:len(l)-2]
                                l=f.readline()
                        row = row + '"\n'
                        outf.write(row)
                        row=""

f.close()
outf.close()
return
```

4) Evidence Code de GO

GO Evidence Code Decision Tree