Rapport final détaillé sur le *workpackage* 2 du projet Abliss

Denis Maurel

1 Description préliminaire

1.1 Rappel du projet

This leads to our second scientific challenge: how to feed such a method with experiments formalized by means of relations? Indeed, if we want the method to be useful, these predicates have to be extracted and formalized automatically from scientific papers. Natural language processing has been used in many medical or biomedical projects, mostly in clinical and consumer-generated text, but also in text mining, with a multiplication of challenge tasks. Our work is situated more precisely in Literature-based discovery. Most projects rely on co-occurrence to predict functional relations between proteins and/or genes. However, only 30 % of protein pairs detected that way are really in interaction. Moreover, co-occurrence only detects very simple binary events. This is principally due to the whole text search.

In Abliss, we decided to limit the analysis to the *Results* part on scientific papers. Indeed, in the introduction, many of the related relations are not those demonstrated in the paper, but results coming from previous works. In the "Discussion" part, authors interpret their results, and it is often difficult to distinguish actual results from speculations. Moreover, not all the experimental results are discussed. We will use CasSys, a system allowing creating cascades of transducers. The first step is to isolate the sentences relating experimental results; the second one is to generate predicates from isolated sentences.

Preliminary work has already shown that it was possible, using CasSys, to extract from a scientific paper, the sentences giving experimental results with good precision and recall. We have also show that cascades can be used to build predicates from these extracted sentences, thus leading to a preliminary predicate dictionary.

Based on this preliminary work, the aim of WP2 will be:

- To write the cascades for building all necessary predicates.
- To evaluate, from a large number of examples, what are the pieces of information that cannot be translated into predicates, because no corresponding predicate exists. And in that case complete the predicate dictionary.
- To implement co-reference methods in order to resolve as much as possible the incomplete predicates.

1.2 La modélisation biologique

Pour permettre des raisonnements automatiques sur les faits découverts dans les articles, il est nécessaire que le formalisme que nous utilisons pour décrire les prédicats soit précis, mais il doit être aussi évidemment adapté au Tal. Quatre points sont, à notre sens, absolument nécessaires :

1. les prédicats et les règles doivent être lisibles à la fois par la machine et par l'humain[1] ;
2. la description doit être chimiquement précise, par exemple, concernant les différents états de la même molécule[2] ;
3. les actions directes ou indirectes doivent être distinguées ;
4. la formulation d'hypothèses doit être possible et celles-ci doivent pouvoir être exprimées dans le même formalisme.

Comme nous l'avons déjà dit, ce n'était pas le cas du formalisme mis en œuvre dans les travaux préliminaires réalisés sur des extractions manuelles. Ce n'était pas le cas non plus des différents formalismes existant par ailleurs au début de notre projet, comme, par exemple, ceux basés sur les réseaux de Petri (Koch, 2010) ou sur les modèles logiques (Morris et al., 2010) ; ou encore d'autres, finalement désignés par les termes d'*executable biology* ou d'*algorithmic systems biology* (Konur, 2020) (Fisher, Henzinger, 2007) (Priami, 2009).

Nous avons donc conçu un nouveau formalisme[3] où les prédicats sont de trois types[4] :

1. les prédicats ontologiques, qui correspondent aux types des composants (gènes, molécules, protéines...)[5] ;
2. les prédicats de relation, qui indiquent les relations entre les composants, etc.[6] ;
3. les prédicats d'action, qui décrivent le ou les effets d'un composant sur un autre[7].

La liste complète de ces prédicats se trouve dans le fichier *predicatsAbliss.xlsx* joint.

Une fois un prédicat d'action détecté, nous lui attribuons un statut en fonction du contexte. Le statut attribué est *confirmed*[8], si nous avons une preuve expérimentale que le fait est vrai, c'est-à-dire si l'expérience qui permet de conclure à ce fait est décrite dans le texte.

Les prédicats instanciés par leurs arguments sont appelés des faits. C'est le recensement de ces faits qui fera l'objet de ce *workpackage*.

1.3 Le corpus

Les articles scientifiques dans les domaines concernant la biologie et la santé suivent pour la plupart un plan précis en cinq points :

1. *Abstract* : le résumé qui reprend les principaux résultats ;
2. *Introduction* : l'introduction qui permet de comprendre le contexte de l'étude et présente l'état de l'art ;
3. *Materials and Methods* : la description du matériel et des méthodes qui seront utilisés dans les expériences ;
4. *Results* : les résultats qui listent et détaillent les expériences réalisées ;

---

[1] En effet, un biologiste doit être capable de comprendre les raisonnements impliqués dans une déduction automatique.

[2] Ainsi, une activation sera décrite comme le passage d'un état *inactif* à un état *actif*.

[3] Nous avons choisi pour cela d'utiliser l'ontologie SBO (*Systems Biology Ontology*) (Novère, 2006), même si nous n'en utilisons qu'une petite partie, celle concernant les réseaux de signalisation.

[4] Les deux premiers types sont regroupés sous le sous-type *background* et on attribue au dernier le sous-type *network*. Cette distinction est utile dans la partie déductive de notre travail, mais ne sera pas utilisée ici.

[5] Il s'agit de dresser la liste des composants du système.

[6] Il s'agit de déterminer les différents états de chaque composant et les actions déclenchées par chaque état.

[7] Comment un état donné d'un premier composant peut déclencher une transition d'états d'un second composant.

[8] Ce qui est majoritairement le cas, puisque nous travaillons sur des résultats d'expériences.

5. *Discussion* : la discussion qui donne une interprétation des résultats ainsi que des perspectives de travail.

Deux remarques importantes sur notre projet :

- Tout d'abord, nous ne nous intéressons pas spécifiquement au sujet de l'article, mais aux différents résultats intermédiaires obtenus lors des expériences. Ces résultats ne figurent en général, ni dans le titre, ni dans le résumé. C'est la combinaison de ces différents résultats sur un ensemble conséquent d'articles qui devrait nous permettre de déceler de nouvelles pistes de recherche concernant les réseaux de signalisation.
- Nous limiterons donc l'extraction de nos faits à la partie *Results* des articles analysés. Cette partie contient majoritairement des données factuelles, réduisant les suggestions et interprétations non démontrées des auteurs ou la référence à des travaux antérieurs.

Le domaine d'application choisi pour le développement de cette méthode est la signalisation des GPCR restreinte à la partie dépendant de β-arrestine et ERK. Ce choix repose sur trois raisons : la reconnaissance internationale d'un des laboratoires du projet sur le domaine de la signalisation GPCR, notamment pour ses études sur le rôle des β-arrestines ; la faisabilité de l'étude par la sélection d'un nombre raisonnable d'articles pour notre corpus ; l'intérêt de cette recherche, car le rôle exact de la protéine ERK est mal connu.

Ceci nous a conduit à la recherche d'articles (disponibles au format XML) à partir de trois mots-clés dans le texte intégral, deux protéines (*arrestin* et *ERK*) avec un prédicat (*phosphorylation*). Nous avons obtenu 5 141 articles les contenant. Parmi ceux-ci, 4 232 ont pu être téléchargés correctement et 435 ont une partie *Results* contenant elle aussi les trois mots-clés. Ce sont ces 435 articles qui ont été analysés. Ils proviennent de 140 journaux différents. La plupart ne sont pas en Open Access et nous sommes dans l'impossibilité de diffuser librement le corpus et les résultats. La version jointe est donc strictement réservée à l'ANR et à l'évaluation du projet.

Notons que nous n'avons pas utilisé de mot-clef relatif aux GPCR car, d'une part, il existe un grand nombre de mots-clefs correspondants, rendant le choix difficile ; et, d'autre part, les articles parlant d'arrestine et de ERK sont dans leur très grande majorité relatifs à des GPCRs.

1.4 Unitex

Nous avons choisi de décrire les faits et de les extraire à l'aide de la plateforme logicielle Unitex (Paumier, 2003). Nous utilisons pour cela des cascades à nombre fini d'états (Abney, 1996), (Hobbs and al., 1996), (Friburger and Maurel, 2004). Ces cascades sont composées de réseaux de transitions enrichies (ATN), présentés par Unitex sous la forme de graphes. Ces graphes utilisent des dictionnaires gérés par la plateforme. Ils ont été tout d'abord écrits à partir de l'étude de onze articles hors corpus, puis complétés et corrigés au fur et à mesure de l'avancement du projet.

Unitex comporte un système interne de scripts qui nous permet de programmer notre chaîne de traitement sous la forme d'un script global où s'enchaînent des scripts PHP et des scripts Unitex.

1.5 Création de dictionnaires

Une des richesses de la plateforme Unitex est l'existence de ressources dictionnairiques et d'outils de construction de dictionnaires spécifiques. Nous avons utilisé le dictionnaire anglais de la distribution afin de reconnaître les verbes, les adjectifs, les prépositions, etc. Nous avons aussi constitué, à partir des bases de données publiques, des listes de gènes et de protéines que nous avons transformées en dictionnaire Unitex. Ces dictionnaires contiennent respectivement 582 372 et 115 007 entrées. Par exemple, la protéine *advillin* figure dans la base utilisée et est associée au symbole officiel *AVIL* ; elle correspondra à deux entrées de notre dictionnaire de

protéines : *advillin,AVIL.cProtein:s* et *advillins,AVIL.cProtein:p*. Le symbole associé devient une forme vedette ; la catégorie est *cProtein* ; la forme est au singulier ou au pluriel. Un graphe peut reconnaître une de ces formes, mais aussi les deux à la fois par le code <*AVIL*> ou même n'importe quelle protéine par le code <*cProtein*>.

Nous avons aussi construit manuellement un dictionnaire complémentaire de 4 193 entrées, correspondant au domaine spécifique des réseaux biochimiques. Ce dictionnaire contient entre autres :

- 1 048 méthodes ;
- 79 familles de molécules ;
- 376 molécules ;
- 195 localisations, correspondant à des compartiments sous-cellulaires ;
- 1 668 organismes ;
- 154 processus biologiques ;
- 373 régulations[9] (effets d'un composant sur un autre).

Pour la partie biologique, l'ensemble de ces dictionnaires correspond à quarante-trois catégories (*cProtein, cGene, cMethods*...).

Restait un problème : les noms de gènes sont extrêmement ambigus avec des mots très courants du vocabulaire anglais ; par exemple *for*, *do* ou *if* sont des gènes ! Pour le résoudre et pour faciliter aussi certains étiquetages en gène ou protéine, nous avons choisi d'utiliser le système de priorité des dictionnaires proposé par Unitex. Celui-ci permet trois niveaux. Ce qu'un dictionnaire d'un niveau inférieur étiquette ne sera pas étiqueté par un dictionnaire de niveau supérieur. Nous avons donc passé nos dictionnaires de la manière suivante (par ordre de priorité) :

1. le dictionnaire des ambiguïtés entre les gènes ou protéines et le vocabulaire anglais courant ;
2. notre dictionnaire spécifique au projet, ainsi que le dictionnaire de gènes et celui des protéines ;
3. le dictionnaire d'anglais fourni par la distribution d'Unitex.

Ces dictionnaires se trouvent dans l'archive *Abliss_lingpkg.zip* jointe, elle-même contenue dans l'archive *scripts.zip*.

1.6 Références

Abney S., « Partial Parsing via Finite-State Cascades », Workshop on Robust Parsing, 8th European Summer School in Logic, Language and Information, Prague, Czech Republic, p. 8-15, 1996.

Fisher J., Henzinger T., « Executable cell biology », Nature biotechnology, vol. 25, no 11, p. 1239-1249, 2007.

Friburger N., Maurel D., « Finite-state transducer cascade to extract named entities in texts », Theoretical Computer Science, vol. 313, p. 94-104, 2004.

Hobbs J., Appelt D., Bear J., Israel D., Kameyama M., Stickel M., Tyson M., « A Cascaded Finite-State Transducer for Extracting Information from Natural-Language Text in Finite State

[9] Ce sont principalement : soit des verbes indiquant une inhibition (*block*, *inhibit*, *attenuate*, *decrease*), une activation (*activation*, *stimulate*, *increase*) ou une expression (*express*, *transcript*, *detectable*, *level*) ; soit des verbes comme *bind*, *block*, *cause*, etc.

Devices for Natural Language Processing », MIT Press, Cambridge, Massachusetts. p. 383-406, 1996.

Koch I., « Petri nets – a mathematical formalism to analyze chemical reaction networks », Molecular Informatics, vol. 29, no 12, p. 838-843, 2010.

Konur S., « A review of modelling and verification approaches for computational biology », 2020.

Novère N. L., « Model storage, exchange and integration », BMC neuroscience, vol. 7, no 1, p. 1-9, 2006.

Morris M., Saez-Rodriguez J., Sorger P., Lauffenburger D., « Logic-based models for the analysis of cell signaling networks », Biochemistry, vol. 49, no 15, p. 3216-3224, 2010.

Paumier S., De la Reconnaissance de Formes Linguistiques à l'Analyse Syntaxique, Thèse de doctorat en Informatique, Université de Marne-la-Vallée, 2003.

Priami C., « Algorithmic systems biology », Communications of the ACM, vol. 52, no 5, p. 80-88, 2009.

2 Présentation détaillée des scripts, des cascades et des graphes réalisés

L'ensemble du WP2 comprend cinq scripts PHP qui appellent six cascades Unitex.

2.1 *Le script PHP* 1PartieResults

Le premier script prend en entrée les fichiers du dossier *0TelechargementNCBI\CorpusNCBI*, extrait la partie *Results*, quand elle existe, et la place dans le dossier *1PartieResultats\ResultatsGlobaux*. Si la partie *Results* n'existe pas, le script crée un fichier contenant juste l'information *No Results Section*. Puis, pour les fichiers contenant une partie *Results*, le script vérifie la présence (dans cette partie) des trois mots clés. Si c'est le cas, le fichier est copié dans le dossier *1PartieResultats\SelectionDesResultats* et le fichier de départ, contenant l'article en entier, est placé dans le dossier *3Entetes/SelectionCorpusNCBI*, pour traitement ultérieur par le troisième script.

Dans l'exemple joint, le dossier *CorpusNCBI* contient quatre fichiers, les articles *84122.txt*, *218003.txt*, *2216744.txt* et *4123065.txt*. Le nom de ces articles est leur numéro *PMCID*. Dans le dossier *ResultatsGlobaux*, le fichier *218003.txt* contient la phrase *No Results Section* et les quatre autres la partie *Results*. Le fichier *84122.txt* ne contient plus les trois mots clés, donc le dossier *SelectionDesResultats* ne contient plus que deux fichiers.

2.1.1 La cascade 1PartieResults

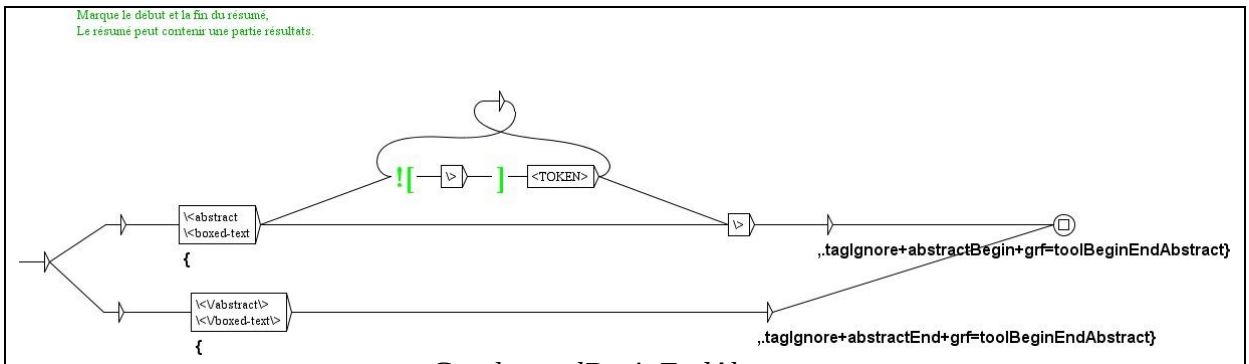
Cette première cascade Unitex est composée de dix-sept graphes.

#	Disabled	Name	Merge	Replace	Until Fix Point	Generaliz...
1	☐	toolBeginEndAbstract.fst2	☑	☐	☐	☐
2	☐	toolEraseAbstract.fst2	☐	☑	☑	☐
3	☐	toolBeginEndResult.fst2	☑	☐	☐	☐
4	☐	toolEraseExceptResult.fst2	☐	☑	☑	☐
5	☐	toolNormalizeCodeHtml.fst2	☑	☐	☐	☐
6	☐	toolEraseNote.fst2	☐	☑	☑	☐
7	☐	toolBeginEndFigure.fst2	☑	☐	☐	☐
8	☐	toolXml.fst2	☑	☐	☐	☐
9	☐	toolEraseFigure.fst2	☐	☑	☑	☐
10	☐	toolEraseTable.fst2	☐	☑	☑	☐
11	☐	toolIgnoreSomeTags.fst2	☑	☐	☐	☐
12	☐	toolParagraph.fst2	☐	☑	☐	☐
13	☐	toolEraseBetweenParagraphs.fst2	☐	☑	☑	☐
14	☐	toolEraseXml.fst2	☐	☑	☑	☐
15	☐	toolEraseTagIgnore.fst2	☐	☑	☐	☐
16	☐	toolEraseBeginEnd.fst2	☐	☑	☐	☐
17	☐	toolReference.fst2	☐	☑	☐	☐

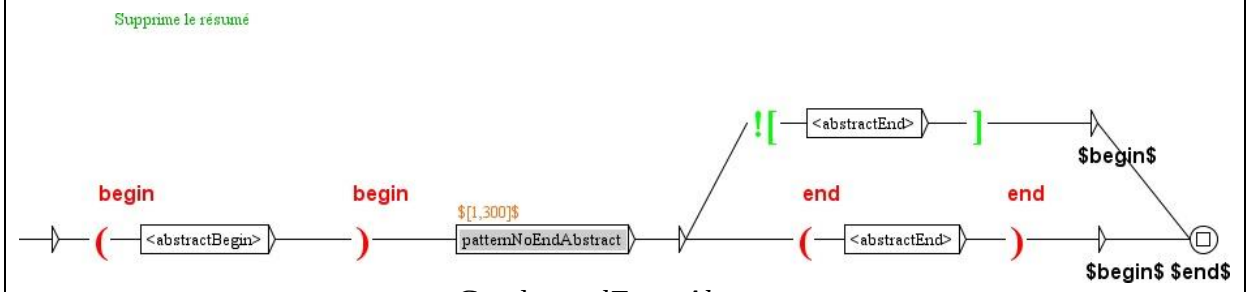
Cascade *1PartieResults*

2.1.1.1 *Les graphes d'effacement*

Les graphes *BeginEnd* marque (si besoin) les parties que les graphes *Erase* suppriment. Une partie *Results* se trouve parfois dans le résumé, nous avons donc choisi de commencer par supprimer celui-ci en passant successivement les graphes *toolBeginEndAbstract* et *toolEraseAbstract*.



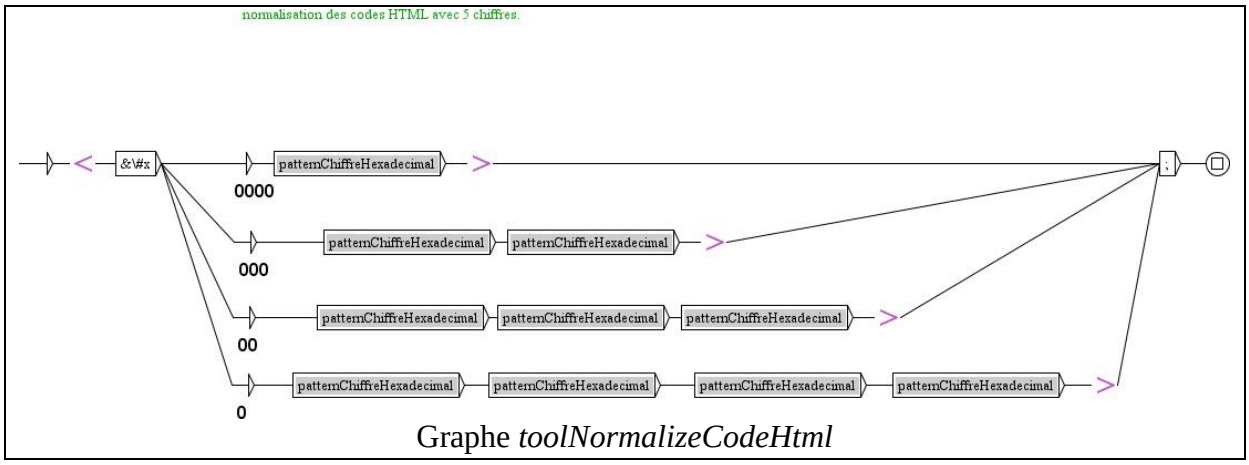
Graphe *toolBeginEndAbstract*



Graphe *toolEraseAbstract*

2.1.1.2 *La normalisation des codes HTML*

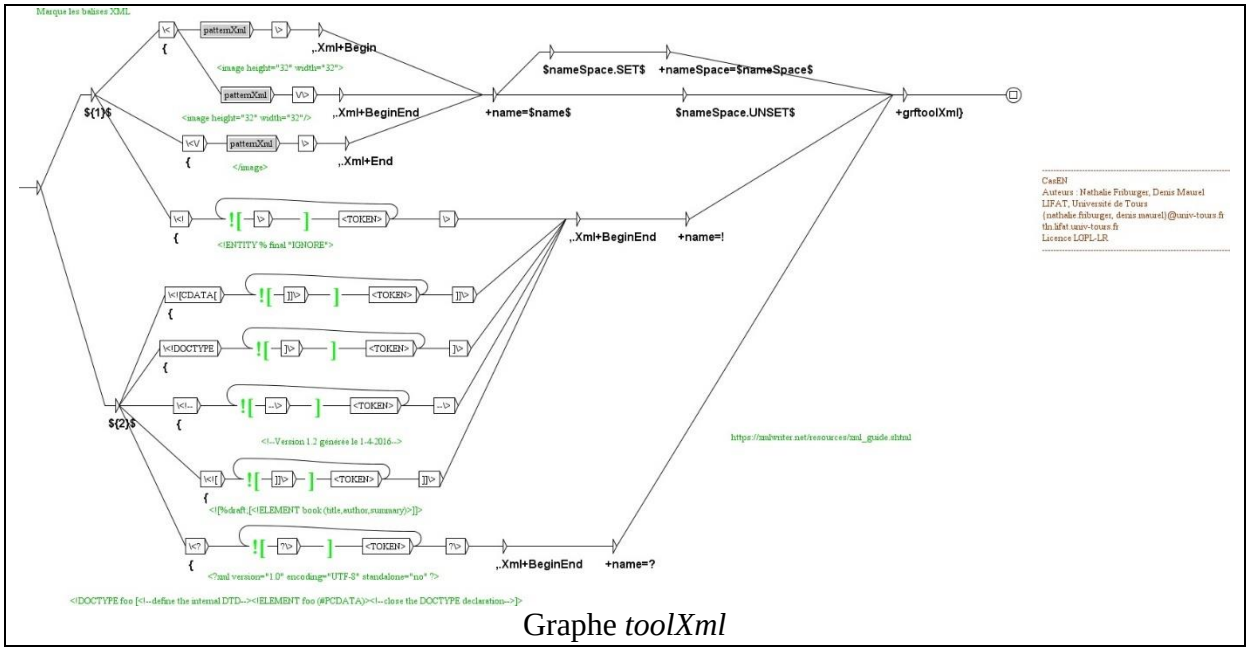
Le graphe *toolNormalizeCodeHtml* normalise les codes HTML sous un format à cinq chiffres hexadécimaux. Ces codes normalisés seront traités en préliminaire à la cascade d'analyse.



Graphe *toolNormalizeCodeHtml*

2.1.1.3 *La reconnaissance des balises XML et des références*

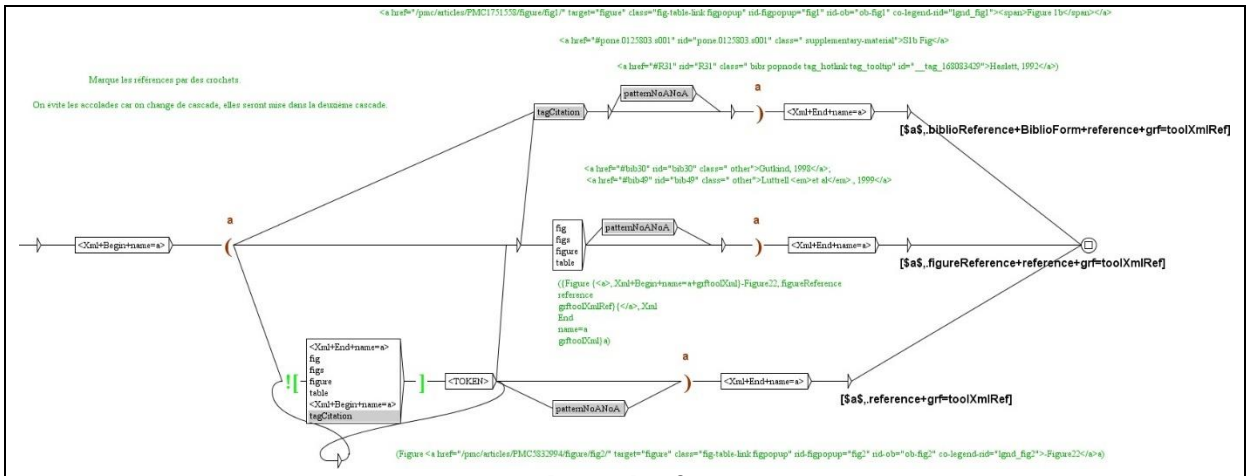
Le graphe *toolXml*, réalisé dans le cadre du projet CasEN[10], reconnaît les balises XML, en gardant accessible le type de balises et son nom.



Graphe *toolXml*

Enfin, le graphe *toolReference* marque les références qui vont être gardées, bibliographie, figures ou autres.

[10] https://tln.lifat.univ-tours.fr/version-francaise/ressources/casen



Graphe *toolReference*

2.2 *Script* 2ResultatsDuCorpus

Le deuxième script lance trois cascades Unitex, respectivement pour l'analyse, la synthèse et la mise en forme des fichiers. Les fichiers traités sont ceux précédemment déposés dans le dossier *1PartieResultats\SelectionDesResultats*. Les fichiers finaux, mis en forme, sont déposés dans le dossier *2ResultatsDuCorpus\ResultatsGlobaux*. Ces fichiers comportent des balises <CRLF/> et <TAB/> pour désigner respectivement un saut de ligne et une tabulation.

Dans l'exemple joint, le dossier *ResultatsGlobaux* contient les deux fichiers, *2216744.result.txt* et *4123065.result.txt*.

2.2.1 La cascade 2_1Analyse

La deuxième cascade Unitex, qui analyse la partie *Results*, est composée de quatorze graphes.

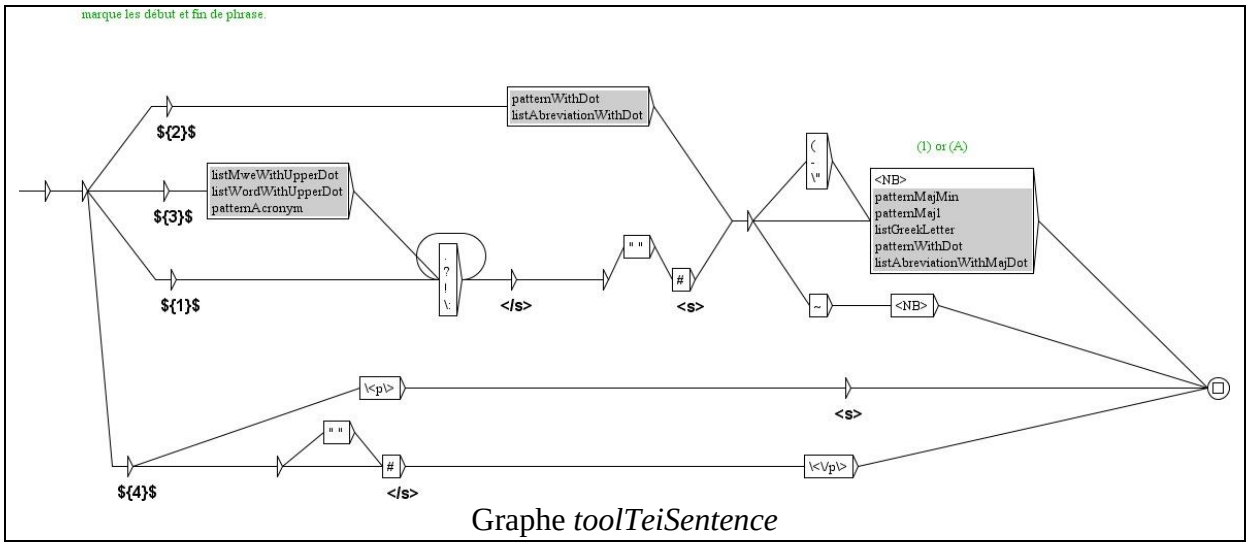
#	Disabled	Name	Merge	Replace	Until Fix Point	Generaliz...
1	☐	toolTeiSentence.fst2	☑	☐	☐	☐
2	☐	toolSpaceAndSerie.fst2	☐	☑	☐	☐
3	☐	toolDescribeExperiment.fst2	☑	☐	☐	☐
4	☐	entity.fst2	☑	☐	☐	☐
5	☐	GeneProtein.fst2	☑	☐	☐	☐
6	☐	ambiguousGeneProtein.fst2	☑	☐	☐	☐
7	☐	extendedEntity.fst2	☑	☐	☐	☐
8	☐	entity.fst2	☑	☐	☐	☐
9	☐	complexEntity.fst2	☑	☐	☐	☐
10	☐	statutHypotesisBiblio.fst2	☑	☐	☐	☐
11	☐	predicatsSimples.fst2	☑	☐	☐	☐
12	☐	verb.fst2	☑	☐	☐	☐
13	☐	predicatsComplexes.fst2	☑	☐	☐	☐
14	☐	networkPredicatComplexImbric.fst2	☑	☐	☐	☐

Cascade *2_1Analyse*

2.2.1.1 Quelques outils préliminaires

Le graphe *toolTeiSentence* provient du projet Istex[11] et a été adapté et amélioré. Il segmente le texte en phrase, balisée par les balises TEI *<s>* et *</s>*.

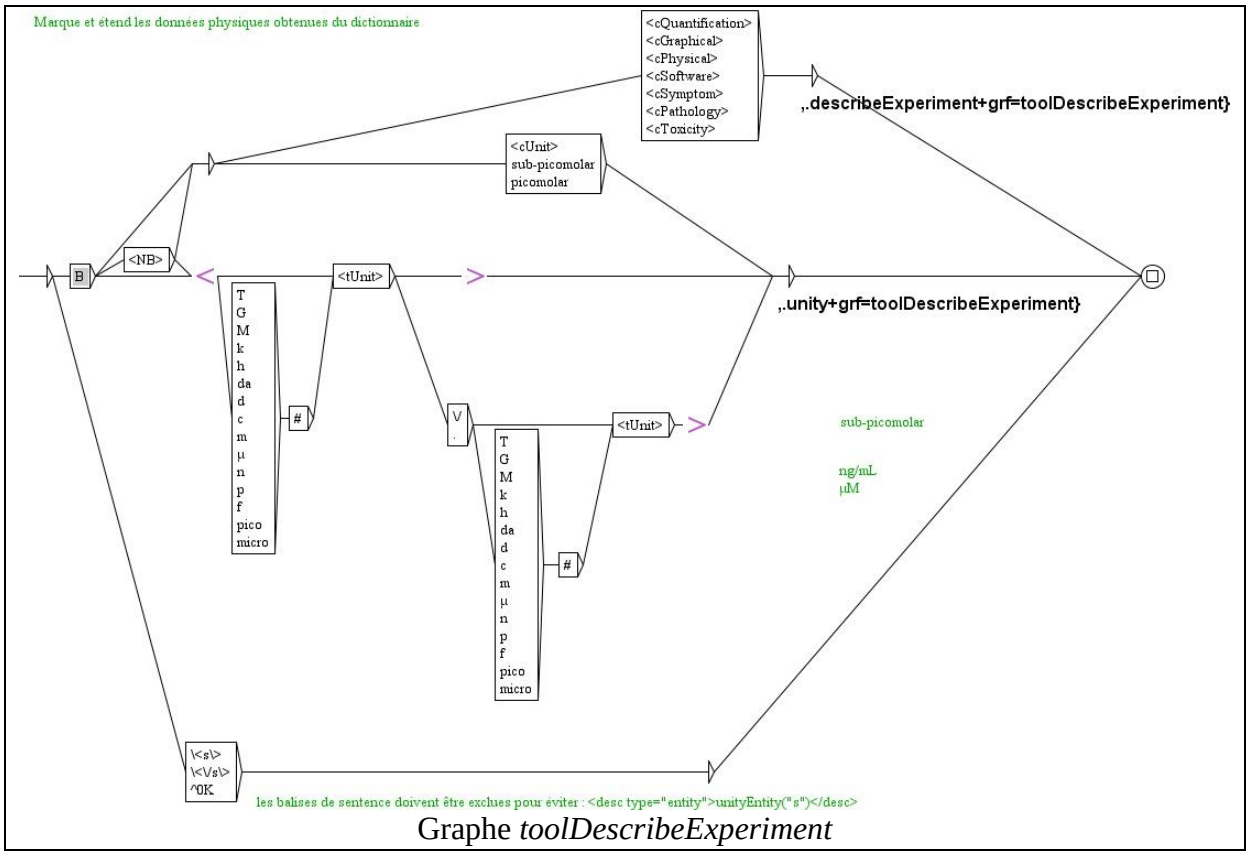
[11] https://tln.lifat.univ-tours.fr/version-francaise/anciens-projets/istex



Graphe *toolTeiSentence*

Le graphe suivant, *toolSpaceAndSerie*, compile trois petites actions : supprimer les espaces entre les phrases, développer des séries[12] et annoter (au sens Unitex) les références[13].

Le graphe *toolDescribeExperiment* permet d'annoter les unités et les descriptions d'expériences obtenues à partir des dictionnaires.



Graphe *toolDescribeExperiment*

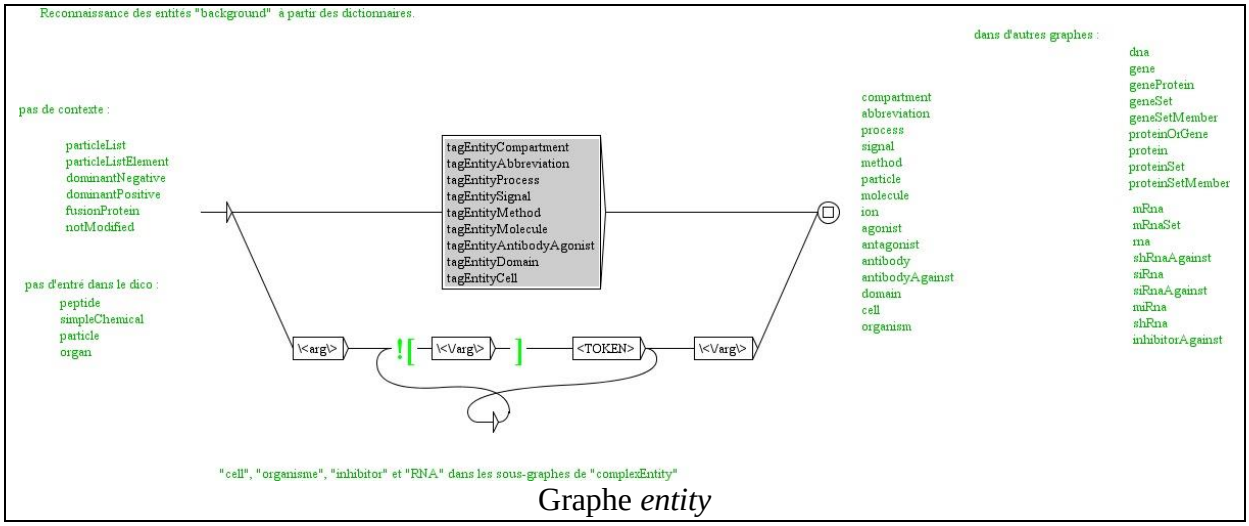
2.2.1.2 La reconnaissance des prédicats ontologiques

Le graphe *entity* reconnaît les entités *background* (les prédicats ontologiques), à partir des dictionnaires. Il serait à compléter. À ce jour, sont reconnues les entités : *compartment*,

[12] Par exemple, la séquence *S1P1 to S1P3* sera réécrite *S1P1, S1P2, S1P3.*

[13] Les références ont déjà été reconnues par la cascade précédente.

abbreviation, *process*, *signal*, *method*, *particle*, *molecule*, *ion*, *agonist*, *antagonist*, *antibody*, *antibodyAgainst*, *domain*, *cell* et *organism*.



Graphe *entity*

Le graphe suivant, *GeneProtein*, reconnaît à partir des dictionnaires et du contexte local, les gènes et les protéines. Il est immédiatement suivi d'un graphe, de désambigüation entre ces deux entités qui portent souvent les mêmes noms, le graphe *ambiguousGeneProtein*. Suivent ensuite la reconnaissance des entités étendues[14], graphe *extendedEntity*, et des entités complexes[15], graphe *complexEntity*.

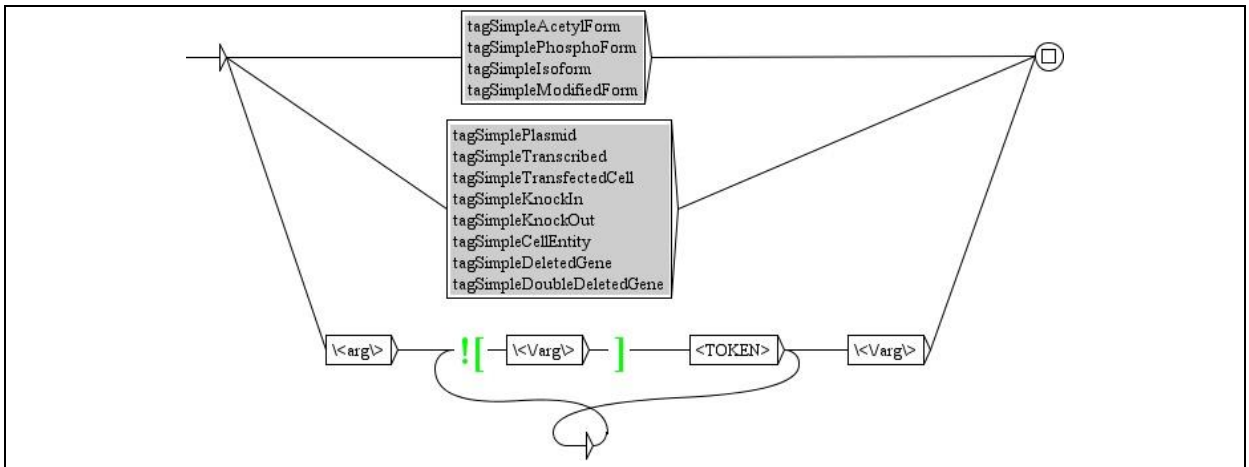
Comme le but du projet Abliss est de décrire des résultats d'expériences réalisées dans le cadre de l'article analysé, le graphe suivant, *statutHypotesisBiblio*, annote les contextes indiquant que l'auteur se réfère à une publication existante ou à une hypothèse. Ce graphe sera certainement à compléter lui aussi, au fur et à mesure de l'analyse de nouveaux articles.

2.2.1.3 La reconnaissance des prédicats de relation

Sur les trente-quatre prédicats de relation actuellement décrits par notre formalisme, nous en avons implantés douze: *acetylForm*, *cellEntity*, *deletedGene*, *doubleDeletedGene*, *isoform*, *knockIn*, *knockOut*, *modifiedForm*, *phosphoForm*, *plasmid*, *transcribed* et *transfectedCell*. Les graphes correspondants annotent les faits, c'est-à-dire le prédicat et sa liste d'arguments, placée entre balises *<arg></arg>*.

[14] Comme *β-Arrestin 2*.

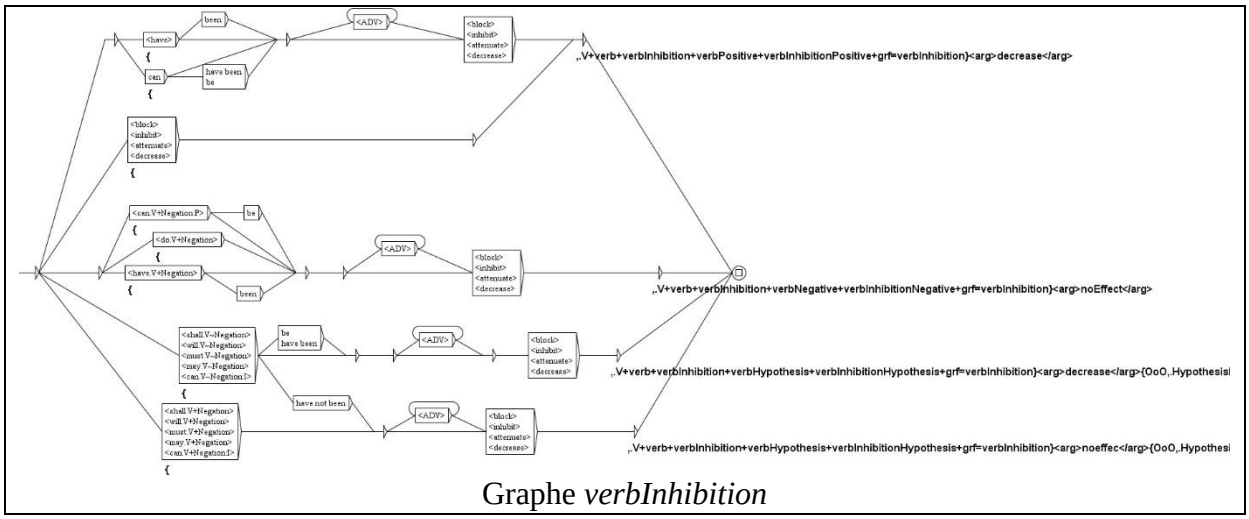
[15] *mRna*, *mRnaSet*, *rna*, *shRnaAgainst*, *siRna*, *siRnaAgainst*, *miRna*, *shRna* et *inhibitorAgainst*.



Graphe *predicatsSimples*

2.2.1.4 La reconnaissance des verbes

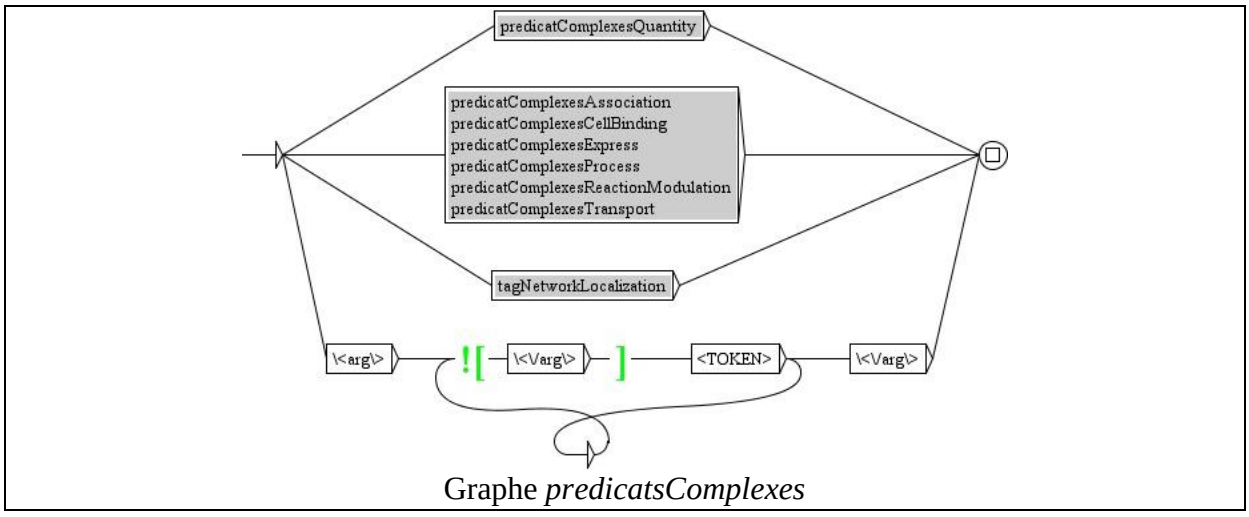
Certains verbes apportent des indications sur la réalisation (ou la non-réalisation) des expériences. Ces verbes, lorsqu'ils sont précédés d'une négation ou d'une forme modale changent les traits du prédicat, respectivement en *noEffect* et *hypothesys*. Certains verbes proviennent du dictionnaire Abliss, d'autres sont regroupés en verbes d'expression, d'inhibition, de stimulation et de différence.



Graphe *verbInhibition*

2.2.1.5 La reconnaissance des prédicats d'action

Les deux derniers graphes de la cascade d'analyse permettent la reconnaissance des prédicats d'action. Notre formalisme en prévoit aujourd'hui cinquante-quatre, mais notre implantation actuelle n'en comporte que vingt-et-un : *association*, *associationCell*, *associationCompetition*, *associationModulation*, *associationModulationPerturbator*, *associationModulationSignal-Compare*, *cellBindingModulation*, *expressed*, *expressedCell*, *expressedCorrelation*, *process-Modulation*, *processModulationPerturbator*, *processTargetedCell*, *quantityModulation*, *reactionModulation*, *reactionModulation-Cell*, *reactionModulationMediation*, *reaction-ModulationPerturbator*, *transportModulation* et *transportModulationPerturbator*. Comme pour les prédicats de relation, les graphes correspondants annotent les faits (prédicat et liste d'arguments, entre balises *<arg></arg>*).



Graphe *predicatsComplexes*

2.2.2 La cascade 2_2Synthese

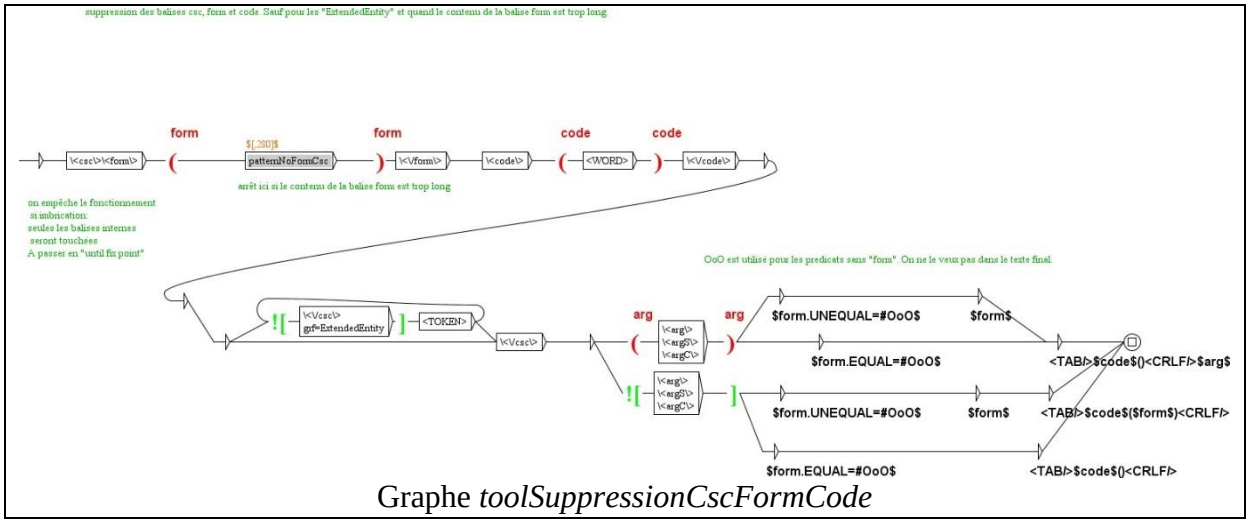
La troisième cascade Unitex fait une synthèse de la cascade précédente, c'est-à-dire qu'elle remplace le balisage Unitex par la présentation qui nous intéresse. Elle est composée de neuf graphes.

#	Disabled	Name	Merge	Replace	Until Fix Point	Generaliz...
1	☐	toolSuppressionCscFormCode.fst2	☐	☑	☑	☐
2	☐	toolSuppressionCscFormCodeExtended.fst2	☐	☑	☐	☐
3	☐	toolSuppressionLargeCscFormCode.fst2	☐	☑	☑	☐
4	☐	toolEntityCleaning.fst2	☐	☑	☑	☐
5	☐	toolBalisesArgCleaning.fst2	☐	☑	☑	☐
6	☐	toolShapePredicatArg.fst2	☐	☑	☐	☐
7	☐	tagMobile.fst2	☑	☐	☐	☐
8	☐	toolDeplaceMobile.fst2	☐	☑	☑	☐
9	☐	toolDeplaceMobileStatut.fst2	☐	☑	☑	☐

Cascade *2_2Synthese*

2.2.2.1 Les graphes de suppression des codes Unitex

Les trois premiers graphes de la cascade sont des graphes de remplacement des codes Unitex. Ces graphes ajoutent aussi après chaque prédicat une tabulation, le nom du prédicat, des parenthèses, un saut de ligne et, éventuellement les arguments du prédicat[16]. Ces graphes sont complétés de deux graphes chargés du nettoyage des entités et des arguments.

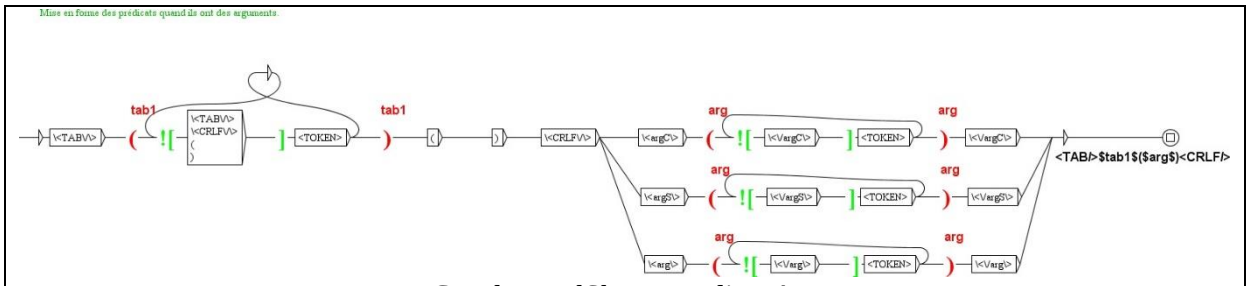


Graphe *toolSuppressionCscFormCode*

[16] Ou, plutôt, les balises <TAB/> et <CRLF/> qui seront au final remplacées respectivement par une tabulation et un saut de ligne. Les arguments ont été placés entre balises *<arg></arg>* par les graphes de la cascade d'analyse.

2.2.2.2 Les graphes de déplacement des faits

Une fois balisés comme indiqué ci-dessus par une tabulation et un saut de ligne, les faits (prédicats suivis d'arguments) sont transformés sous la forme classique, *predicat(x, y, z)*. Puis tous les faits sont déplacés après la fin de phrase, c'est-à-dire entre la fin de la phrase et la phrase suivante du paragraphe (entre les balises *</s>* et *<s>*), ou entre la fin de la phrase et la fin du paragraphe (entre les balises *</s>* et *</p>*). Ce déplacement est effectué après avoir balisé les éléments à déplacer avec des balises Unitex.



Graphe *toolShapePredicatArg*

2.2.3 La cascade 2_3MiseEnForme

La quatrième cascade Unitex est chargée de la mise en forme finale du fichier de résultats. Elle est composée de six graphes.

#	Disabled	Name	Merge	Replace	Until Fix Point	Generaliz...
1	☐	toolSuppressionCscFormCodeMobile.fst2	☐	☑	☐	☐
2	☐	toolMergeStatutHypothesis.fst2	☑	☐	☐	☐
3	☐	toolMergeStatutBiblio.fst2	☑	☐	☐	☐
4	☐	toolReplaceStatut.fst2	☐	☑	☐	☐
5	☐	toolSuppressionUselessPredicate.fst2	☐	☑	☑	☐
6	☐	toolDescDiv.fst2	☐	☑	☐	☐

Cascade *2_3*MiseEnForme

Le premier graphe de la cascade supprime les balises Unitex qui ont servi au déplacement des faits dans la cascade précédente. Les trois graphes suivants ajoutent l'attribut indiquant le statut du prédicat de relation (*hypothesis*, *biblio* ou *confirmed*).

Après un graphe de nettoyage, le dernier graphe insère des balises * *, *<sentence></sentence>* et *<paragraph></paragraph>*.

2.3 Le script 3Entetes

Le troisième script lance la cascade Unitex qui va récupérer les fichiers sélectionnés par le premier script, qui les a placés dans le dossier *3Entetes/SelectionCorpusNCBI*. Il en extrait les métadonnées qui seront insérées dans les fichiers finaux. Ces métadonnées sont placées dans le dossier *3Entetes/ResultatsGlobaux*. Elles sont composées des balises *<title>*, *<author>*, *<publisher>*, *<date>*, *<biblScope>* et *<idno>*.

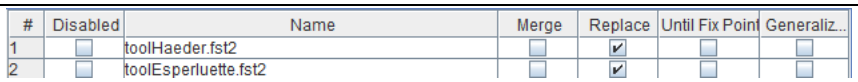
Dans l'exemple joint, le dossier *SelectionCorpusNCBI* et le dossier *ResultatsGlobaux* contiennent les deux fichiers correspondant.

Le dossier *3Entetes* contient aussi les trois fichiers spécifiant les entêtes des trois fichiers résultats : *EnteteDuFichierResults.txt*, *EnteteDuFichierStatistiqueGlobal.txt* et *EnteteDu-FichierStatistiqueTexteParTexte.txt*. Ces trois entêtes présentent le projet Abliss et ne diffèrent que par le contenu des balises *<note></note>*.

2.3.1 La cascade 3Entete

La cascade *3Entete* ne contient que deux graphes outils, le premier récupère l'attribut *name* des balises *meta* pour en faire un nom de balise. Ceci afin de permettre à la fonction *Unxmlize* du

script Unitex *3Entetes* de récupérer les informations souhaitées. Le deuxième graphe convertit les esperluettes éventuellement présentes dans ces balises en une forme textuelle (*and*). Ceci afin de garantir la production finale d'un texte à la norme XML.



#	Disabled	Name	Merge	Replace	Until Fix Point	Generaliz...
1	☐	toolHaeder.fst2	☐	☑	☐	☐
2	☐	toolEsperluette.fst2	☐	☑	☐	☐

Cascade *3Entete*

2.4 Le script 4FichierResults

Le quatrième script est un script PHP qui crée le fichier des résultats globaux dans lequel chaque article sera associé à ces métadonnées.

Dans l'exemple joint, le fichier *results20230623.xml* a la structure ci-dessous.

```
<?xml version="1.0" encoding="UTF-8"?>
<teiCorpus xmlns="http://www.tei-c.org/ns/1.0">
	<teiHeader>
Entête générale du projet Abliss
	</teiHeader>
	<tei>
		<teiHeader>
Entête de l'article PMC2216744
		</teiHeader>
		<text>
Résultats de l'article PMC2216744
			<div type="paragraph">
Premier paragraphe de l'article
				<div type="sentence">
Première phrase de l'article
					<desc type="background" subtype="ontological">
Premier fait de la première phrase
					</desc>
…
				</div>
…
			</div>
…		</text>
	</tei>
	<tei>
		<teiHeader>
Entête de l'article PMC4123065
		</teiHeader>
		<text>
Résultats de l'article PMC4123065
		</text>
	</tei>
</teiCorpus>
```

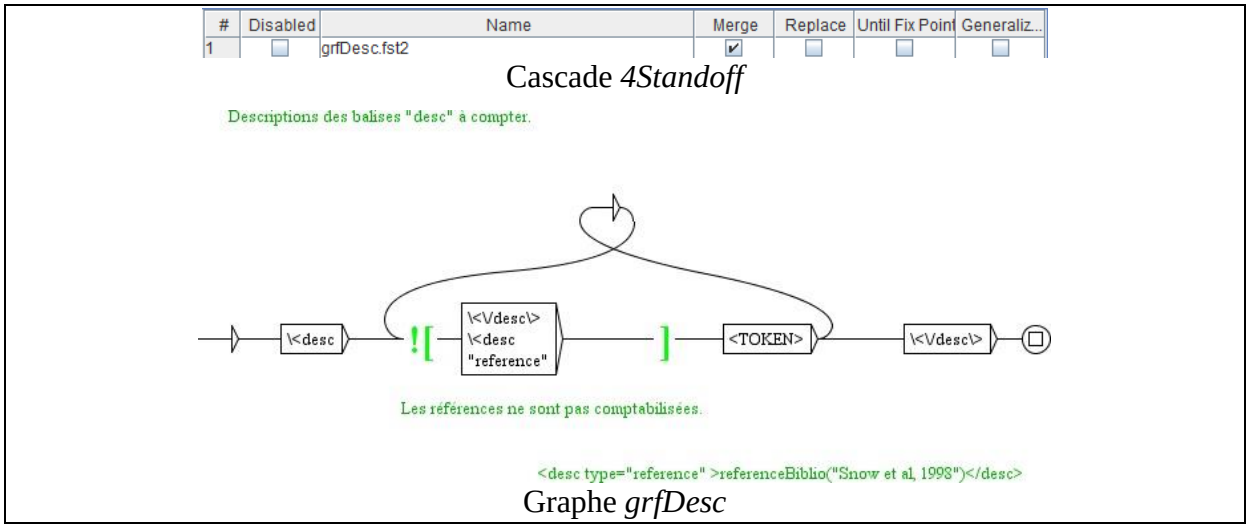
2.5 Le script 5FichiersStatistiques

Le cinquième et dernier script lance une cascade Unitex standoff pour créer les deux fichiers statistiques. Un premier fichier comprenant les occurrences d'apparition des faits sur l'ensemble

de tous les articles et un deuxième fichier comprenant les occurrences d'apparition des faits, article par article.

2.5.1 La cascade 4Standoff

Cette dernière cascade comprend un seul graphe qui extrait le contenu des balises <desc></desc> et les comptabilise. Ce contenu est classé par catégorie de prédicats.



Cascade *4Standoff*

Graphe *grfDesc*

Dans l'exemple joint, le fichier *statistiquesGlobales20230328.xml* a la structure ci-dessous.

<?xml version="1.0" encoding="UTF-8"?>
<teiCorpus xmlns="http://www.tei-c.org/ns/1.0">
	<teiHeader>
Entête générale du projet Abliss
	</teiHeader>
	<listAnnotation tag="desc" subtype="action">
Liste des faits d'action
		<action>
			<fact>
Premier fait d'action
			</fact>
			<frequency value="
Nombre d'occurrences de ce fait
			"/>
		</action>
…
	</listAnnotation>
	<listAnnotation tag="desc" subtype="ontological">
Liste des faits ontologiques
	</listAnnotation>
	<listAnnotation tag="desc" subtype="relation">
Liste des faits de relation
	</listAnnotation>
</tei>

Les scripts PHP de lancement des programmes se trouvent dans l'archive *Scripts.zip*, avec les six fichiers donnés en exemple, et les cascades, graphes et scripts Unitex dans l'archives *Abliss_lingpkg.zip*, elle-même contenue dans l'archive *scripts.zip*.

Les résultats obtenus sur l'ensemble du corpus, au jour de l'évaluation, sont joints dans l'archive *fichiersResultats20230120.zip*. Celle-ci contient les trois fichiers *results20230120.xml*, *statistiquesGlobales20230110.xml* et *statistiquesParArticle20230110.xml*.

3 Évaluation

3.1 Présentation

Pour l'évaluation de notre travail, nous avons collecté auprès de nos collègues biologistes une liste de sept articles postérieurs à la constitution de notre corpus. Ces sept articles étaient disponibles au format XML et possédaient une partie *Result* contenant nos trois mots-clés. Il s'agit de (Luo et al., 2022), (Pani et al., 2021), (Stoddart et al., 2021), (Cong et al., 2021), (Zhang et al., 2021), (Mo et al., 2022) et (Sharma et al., 2021).

Pour les trois premiers papiers, l'évaluation a été faite par deux expertes du domaine, de manière indépendante dans un premier temps. Puis les évaluations ont été confrontées l'une a l'autre. Il est apparu que les deux expertes s'accordaient sur 73,5 % des évaluations. Elles ont alors argumenté leurs évaluations, pour chaque prédicat faisant apparaitre une différence, et se sont accordées sur une évaluation commune ; c'est cette évaluation de consensus qui apparait dans les tableaux. Pour les quatre papiers restants, l'évaluation a été faite par une seule des deux expertes.

Dans les tableaux ci-dessous, nous avons comptabilisé les *Vrais positifs*, les *Faux négatifs* et les *Faux positifs*. Nous n'avons pas pris en compte les *Vrais négatifs*. En effet, il est impossible d'évaluer tous les prédicats non-relevants qui pourraient être trouvés dans une phrase.

Dans un premier temps, nous avons séparé notre évaluation en deux parties distinctes :

1. Celle des prédicats ontologiques (comment sont-ils reconnus et étendus par nos graphes, à partir de nos dictionnaires) :

	Vrais positifs	Faux positifs	Faux négatifs
(Luo et al., 2022)	139	66	2
(Pani et al., 2021)	166	242	10
(Stoddart et al., 2021)	254	51	39
(Cong et al., 2021)	114	287	12
(Zhang et al., 2021)	146	53	6
(Mo et al., 2022)	409	122	18
(Sharma et al., 2021)	328	703	30
Total	**556**	**1524**	**117**
Comptabilisation des prédicats ontologiques			

Précision	Rappel	F-mesure
50,5 %	93 %	65,5 %
Précision et rappel pour les prédicats ontologiques		

2. Celle des prédicats de relation et d'action (l'évaluation de nos graphes spécifiques à ces prédicats) :

	Vrais positifs	Faux positifs	Faux négatifs
(Luo et al., 2022)	15	6	3
(Pani et al., 2021)	6	19	7
(Stoddart et al., 2021)	9	29	9
(Cong et al., 2021)	6	13	10
(Zhang et al., 2021)	9	13	11
(Mo et al., 2022)	36	21	7
(Sharma et al., 2021)	13	31	11
Total	**94**	**132**	**58**
Comptabilisation des prédicats de relation et d'action			

Précision	Rappel	F-mesure
41,6 %	61,8 %	49,7 %
Précision et rappel pour les prédicats de relation et d'action		

Puis, dans un deuxième temps, nous avons refait nos calculs en considérant l'ensemble de nos prédicats :

Précision	Rappel	F-mesure
49,9 %	90,4 %	64,3 %
Précision et rappel pour l'ensemble des prédicats (ontologiques, de relation et d'action)		

L'archive *evaluation.zip* jointe comprend les articles originaux, les fichiers résultats correspondants et l'évaluation détaillée.

3.2 Discussion

Comme cela vient d'être présenté ci-dessus, nous obtenons une F-mesure globale de 64,3 %. Ce qui nous place au-dessus de l'état de l'art et nous encourage vivement à poursuivre nos travaux. Ajoutons quelques remarques issues de cette évaluation.

Il apparait que la source d'erreur la plus importante est la difficulté à distinguer entre gène et protéine. En effet, en biologie, une protéine et le gène qui code pour cette protéine portent le même nom. Ainsi, la présence de ce nom dans le dictionnaire ne suffit pas à identifier la nature de la molécule dont on parle dans une phrase et il est nécessaire de tenir compte du contexte.

Une autre source d'erreurs importante est que le nom des protéines et des gènes est souvent légèrement modifié pour indiquer une propriété particulière. Par exemple, il est courant d'ajouter un *h* ou un *m* devant le nom d'une protéine pour indiquer que l'on parle respectivement de la protéine humaine ou murine (pour la protéine *ERK1*, on utilisera *hERK1* et *mERK1*). Il est également fréquent d'ajouter un *p* pour parler de la version phosphorylée de la protéine, par exemple *pERK1*. Également, on peut citer le fait que, pour parler indifféremment de *ERK1* ou *ERK2*, on utilisera l'expression *ERK1/2*.

Ce type de modification complique la tâche de reconnaissance des entités *gène* et *protéine*, qui sont pourtant essentielles pour la détection et la construction de prédicats de relation et d'action. De plus, si nous avions déjà inclus dans nos cascades les éléments nécessaires au traitement des cas cités ci-dessus, lors du traitement des articles du jeu d'évaluation, nous sommes tombés sur un nouveau cas qui est celui du marquage radioactif d'une molécule. Par exemple, *[3H]-cAMP* représente la molécule *cAMP* marquée au tritium. N'ayant pas rencontré cette syntaxe dans les premiers articles, elle n'est pas prise en compte dans nos cascades.

Parmi les sources d'erreurs on peut également citer l'absence de certains mots dans nos dictionnaires spécialisés. Si on prend l'exemple de la phrase :

> *Treatment with 10 muM NECA for 60 minutes caused a substantial increase of BiFC fluorescence in both cell lines indicating formation and internalization of receptor-beta-arrestin complexes*

nous aurions aimé pouvoir y trouver le prédicat :

> *associatioModulation(NECA,receptor,beta-arrestin,receptor-betaarrestin, increase,uD,confirmed,uC,BiFC)*

indiquant que la molecule *NECA* active l'association entre le récepteur et la β-arrestin. Ce prédicat n'est pas trouvé car le terme *formation* ne figure pas dans notre dictionnaire spécialisé, contrairement au terme *binding*. Ainsi, si le terme employé par les auteurs avait été *binding*, le prédicat aurait été trouvé. Pour remédier à ce type d'erreur, nous ajoutons les mots nécessaires dans nos dictionnaires au fur et à mesure, mais nous anticipons le fait qu'il faudra un nombre important de cycles d'évaluation avant que ces dictionnaires soient réellement proches d'être complets.

Un autre aspect important est la présence dans les textes de nombreuses abréviations. Certaines de ces abréviations sont très génériques, par exemple *ECD* pour *Extracellular Domaine*, et peuvent donc être ajoutées aux dictionnaires. Il est cependant important dans ce cas de bien cadrer leur reconnaissance dans les graphes, car par exemple ECD est également une protéine (*Protein ecdysoneless homolog*). Au contraire, certaines abréviations sont spécifiques d'un texte en particulier, par exemple dans le texte (Cong et al., 2021), *SV1* est utilisé pour désigner un variant de traduction du récepteur *GHRHR*. *SV1* étant également un gène/protéine (mais sans rapport avec le *GHRHR*), les predicats *protein(SV1)* et *gene(SV1)* sont bien construits. Cependant, le lien entre les deux entités est perdu. Ainsi, nous estimons qu'il sera nécessaire de

développer des procédures pour la prise en compte des abréviations spécifiques au texte analyse. Nous anticipons plusieurs difficultés majeures dans ce travail :

- Si certains journaux imposent une section dédiée à la définition des abréviations, ce n'est pas le cas pour tous. Dans le cas où cette information n'est pas présente, il faut détecter dans le texte la définition, et la distinguer d'autres éléments apparaissant entre parenthèses.
- Les abréviations étant dépendantes du texte, leur prise en compte nécessitera :
    1. leur détection au moyen de graphes
    2. la constitution d'un dictionnaire spécifique au texte étudié
    3. la prise en compte de ce dictionnaire avant le passage des graphes décrits dans ce travail.

Enfin, certaines phrases sont très complexes. Par exemple, dans l'un de nos textes d'évaluation, on trouve la phrase

> *For both A3-GFP and A3 R108A-GFP, there was an increase in receptor density (N/mum2), slowing in the diffusion (D, mum2/s), and increase in receptor aggregation (molecular brightness, epsilon) at the upper plasma membrane upon NECA treatment.*

Dans cette phrase, les auteurs rapportent l'observation expérimentale de trois effets différents (la densité de récepteur, la diffusion et l'agrégation) provoqués par une même molécule (*NECA*) sur deux versions différentes du récepteur (*A3-GFP* et *A3 R108-GFP*). Bien que nous parvenions dans cette phrase à identifier les acteurs et les types d'effets, nous ne sommes pas capables de construire tous les prédicats correspondants. Nous pensons que pour ce type de phrases complexes, il sera très difficile de parvenir à un résultat parfait.

On peut également remarquer de grandes variations dans la précision et le rappel entre les différents articles. Par exemple, dans le premier article (Cong et al., 2021), la précision sur les prédicats ontologiques n'est que de 28,4 %, alors qu'elle est de 83,3 % pour le dernier (Stoddart et al., 2021). Ce résultat reflète le fait que le dernier article est très semblable dans la construction, les méthodes utilisées et le type de résultats obtenus, aux articles qui nous ont servi à construire les graphes, alors que le premier aborde des questions et des méthodes que nous n'avions pas encore vues : les biais de signalisation, les variantes de traduction et l'activité constitutive des récepteurs (c'est-à-dire indépendamment du ligand).

3.3 Références des textes évalués

Cong Z., Zhou F., Zhang C., Zou X., Zhang H., Wang Y., Zhou Q., Cai X., Liu Q., Li J., Shao L., Mao C., Wang X., Wu J., Xia T., Zhao L., Jiang H., Zhang Y., Xu H., Cheng X., Yang D., Wang M., « Constitutive signal bias mediated by the human GHRHR splice variant 1 », Proc Natl Acad Sci U S A., Oct, 2021.

Luo J., Pascali F. D., Richmond G., Khojah A., Benovic J., « Characterization of a new WHIM syndrome mutant reveals mechanistic differences in regulation of the chemokine receptor CXCR4 », J Biol Chem., Feb, 2022.

Mo H., Ren Q., Song D., Xu B., Zhou D., Hong X., Hou F., Zhou L., Liu Y., « CXCR4 induced podocyte injury and proteinuria by activating -catenin signaling », Theranostics., vol. 1, no 12(2), p. 767-781, Jan, 2022.

Pani B., Ahn S., Rambarat P., Vege S., Kahsai A., Liu A., Valan B., Staus D., Costa T., Lefkowitz R., « Unique Positive Cooperativity Between the β-Arrestin-Biased-Blocker

Carvedilol and a Small Molecule Positive Allosteric Modulator of the 2-Adrenergic Receptor », Mol Pharmacol., Dec, 2021.

Sharma V., Yang X., Kim S., Mafi A., Saiz-Sanchez D., Villanueva-Anguita P., Xiao L., Inoue A., 3rd W. G., Loh Y., « Novel interaction between neurotrophic factor- 1/carboxypeptidase E and serotonin receptor, 5-HTR1E, protects human neurons against oxidative/neuroexcitotoxic stress via β-arrestin/ERK signaling », Cell Mol Life Sci., Dec, 2021.

Stoddart L., Kilpatrick L., Corriden R., Kellam B., Briddon S., Hill S., « Efficient G protein coupling is not required for agonist-mediated internalization and membrane reorganization of the adenosine A3 receptor », FASEB J., Apr, 2021.

Zhang R., Niu Y., Pan K., Pang H., Chen C., Yip C., Ko W., « β2-Adrenoceptor Activation Stimulates IL-6 Production via PKA, ERK1/2, Src, and Beta-Arrestin2 Signaling Pathways in Human Bronchial Epithelia », Lung., vol. 199, no 6, p. 619-627, Dec, 2021.

4 Conclusion

De nombreuses taches restent à accomplir.

En particulier, nous souhaitions travailler sur les attributs encore non identifiés dans les prédicats de relation et d'action. En effet, les arguments *Distance* (la relation est-elle directe ou indirecte ?), *Cell- Type* et *Method*, qui précisent l'organisme dans lequel l'expérience a été effectuée et la méthode utilisée, sont le plus souvent non instanciés, car inconnus dans le contexte local. L'idée était de faire cette recherche dans le début du même paragraphe, car un résultat expérimental est souvent précédé de la description du contexte et des protocoles expérimentaux. Nous n'avons pas eu le temps d'implanter ce point qui nous semble réalisable par l'insertion dans nos scripts d'une cascade supplémentaire dédiée.

Comme discuté précédemment, la question des abréviations devra également être résolue rapidement, car elle est une source importante d'erreurs. La constitution préalable à l'analyse d'un article d'un dictionnaire des abréviations qu'il contient semble tout à fait réalisable. Nous n'avons cependant pas pu nous y consacrer.

D'autre part, les prédicats définis dans notre modélisation n'ont pas encore tous été implantés dans les cascades, en particulier seuls vingt-et-un prédicats d'action sur les cinquante-quatre définis peuvent actuellement être trouvés dans les textes. Là encore, notre système peut tout à fait être complété par l'ajout de nouveaux graphes dans la cascade d'analyse.

Enfin, comme évoqué à la fin de l'analyse des résultats d'évaluation, les articles utilisés pour construire et affiner les graphes ne sont pas encore assez diversifiés. La prise en compte de nouveaux articles portant sur des processus biologiques différents, par des méthodes différentes, devrait nous permettre d'augmenter encore les performances.

Indépendamment du *workpackage* 1 du projet (*Knowledge-based inference*), notre capacité à construire et instancier les prédicats de manière automatique dans les textes permet déjà l'extraction de l'information biologique dans un très grand nombre d'articles. Ainsi, étant donné que nous avons pris soin de définir notre modèle de manière à ce que les prédicats soient compréhensibles par des humains, le savoir extrait peut d'ores et déjà être utilisé par les biologistes.

Fichiers en annexe

Rappelons que ces fichiers sont joints pour l'évaluation du projet et ne doivent pas être diffusés car la plupart ne sont pas en *open access*.

1. corpusNCBI4232.zip :

L'ensemble du corpus.

2. evaluation.zip :

Les sept articles originaux.

Les fichiers résultats correspondants.

Et l'évaluation détaillée.

3. fichiersResultats20230120.zip :

Les trois fichiers (résultats, statistiques globales et par article) sur l'ensemble du corpus au moment de la dernière évaluation.

4. fichiersResultats20230721.zip :

Les trois fichiers (résultats, statistiques globales et par article) sur l'ensemble du corpus à la fin du projet.

5. predicatsAbliss.xlsx :

La description des prédicats.

6. scripts.zip :

Les scripts PHP

Les fichiers d'exemples

Et le package linguistique.

7. scriptsExempleOpenAcces.zip :

Identique au 6. avec un seul fichier d'exemple (en *open access*).

URL sur le site TLN du Lifat

https://tln.lifat.univ-tours.fr/version-francaise/projets-en-cours/abliss