CORPUS

Corpus

26 | 2025
Bruit de fond ou valeur ajoutée ? Gérer le bruit lors des traitements informatiques des corpus linguistiques

Extraire des données textuelles pour l'analyse du discours : le Détricoteur

Extraction of textual data for discourse analysts: le Détricoteur

Romuald Dalodiere and Manuel Jordan

OpenEdition Journals

Electronic version
URL: https://journals.openedition.org/corpus/9742
DOI: 10.4000/1364y
ISSN: 1765-3126

Publisher
Bases ; corpus et langage - UMR 6039

Provided by Université Côte d'Azur

UNIVERSITÉ CÔTE D'AZUR BIBLIOTHÈQUES

Electronic reference
Romuald Dalodiere and Manuel Jordan, "Extraire des données textuelles pour l'analyse du discours : le Détricoteur", *Corpus* [Online], 26 | 2025, Online since 10 January 2025, connection on 28 January 2025. URL: http://journals.openedition.org/corpus/9742 ; DOI: https://doi.org/10.4000/1364y

Extraire des données textuelles pour l'analyse du discours : le Détricoteur

Extraction of textual data for discourse analysts: le Détricoteur

Romuald Dalodiere and Manuel Jordan

AUTHOR'S NOTE

Une version du Détricoteur est disponible sur demande.

Introduction

1 La généralisation de l'accès au Web par le grand public n'a pas manqué de toucher les linguistes, qui en ont rapidement mesuré le potentiel et démultiplié les usages : l'exploitation des moteurs de recherche comme outils d'exploration linguistique (pour attester de l'existence de structures, relever des fréquences d'occurrences, repérer des usages sociolinguistiques…) s'est étoffée de l'appréhension des modes de communication permis par le Web, ou du genre dont celui-ci est susceptible de relever, par exemple. Dans une perspective plus « technique », le traitement automatique du langage (dorénavant TAL) s'est naturellement montré lui aussi très intéressé par la question du Web, dont il s'est emparé afin de développer de nombreux procédés (voir en particulier Tanguy 2013 : 3-8 pour plus de détails et références associées).

2 Alors que l'aubaine représentée par ce soudain amas de données a conduit les chercheurs à s'interroger sur le sens de la notion de « corpus » et la nécessité de sa redéfinition (*ibid.*, Gatto 2011), une partition s'est imposée entre deux conceptions ; celles du *Web as corpus* et du *Web for corpus* (De Schryver 2002). Dans le *Web as corpus*, l'entièreté du Web est vue comme un corpus potentiel, tandis que dans le *Web for corpus*, la toile n'est qu'un moyen d'accès à des données utilisées pour la constitution de

corpus. De la même façon, des divergences n'ont pas manqué de poindre quant au statut de cette ressource (en particulier dans une conception *Web as corpus*) : Sinclair (2004) refuse au Web le statut de corpus, à cause de ses dimensions inconnues, de son caractère changeant, et de sa constitution hors de toute démarche initialement linguistique. Pour Kilgarriff et Grefenstette (2003 : 333-334), en revanche, la question de ce qui constitue, ou non, un corpus, est fréquemment amalgamée avec celle, sous-jacente, de l'adéquation d'un corpus avec la tâche entreprise : ils plaident pour une conception élargie de la notion de corpus et soutiennent, directement et sans ambages, la pertinence du Web comme corpus.

3 Les spécialistes de la linguistique de corpus et du TAL ne sont pas les seuls à travailler avec des corpus – ni les seuls à trouver un intérêt aux ressources offertes par le Web, par ailleurs. L'analyse du discours (dorénavant AD), située « au carrefour des sciences humaines » (Maingueneau 1996 : 12), travaille moins sur le langage au sens strict, mais s'appuie volontiers sur des collections de texte. C'est dans cette discipline que se situe notre perspective, et plus précisément dans sa mise en œuvre outillée pour l'étude de corpus : la textométrie, ou, si l'on préfère, l'analyse statistique de données textuelles (dorénavant ADT), pour en rester à une dénomination englobante qui fasse fi du foisonnement terminologique (Sitri & Barats 2017a : 11). Le propos de cet article, plus technique que fondamentalement épistémologique, intéressera en premier lieu des chercheurs qui ne sont pas formés à l'utilisation des outils et commandes informatiques propres au TAL (langages de programmation, notamment) : analystes du discours et chercheurs en linguistique de corpus dont les travaux amèneraient à devoir extraire le contenu textuel de sites Web.

4 Le reste de l'article est organisé comme suit : nous commençons par situer les besoins de l'AD en matière de corpus et son positionnement, vis-à-vis du TAL en particulier. Dans un deuxième temps, nous procédons à une brève revue de la littérature, dont l'orientation reste volontairement généraliste afin de correspondre autant que possible aux attentes du public susceptible d'être intéressé par l'outil que nous proposons. Nous présentons ensuite le programme en question, le Détricoteur, avec les détails de sa prise en main, et finissons en indiquant les forces et faiblesses du logiciel ainsi que son champ d'application idéal.

Problématique : des attitudes et des besoins différents

5 Dans le sillage de Kilgarriff et Grefenstette, nous défendons le point de vue selon lequel la question de ce qui constitue un corpus doit être abordée en fonction du linguiste et de son objectif. À cet égard, il semble y avoir une différence d'approche entre le TAL et l'AD. Au sens le plus large, on peut certes envisager la linguistique de corpus dans une acception générale et englobante : Habert *et al.* (1997) parlent d'ailleurs *des* linguistiques de corpus pour souligner la multiplicité de démarches exploitant des corpus. Cette conception ouvre par exemple la voie à des travaux qui, tout en recourant à la dénomination de « linguistique de corpus » et en mobilisant ses outils, se situent en même temps en AD (voir par exemple Alexander 2009, Bernard 2015, Jaworska & Nanda 2016, Fernández-Vázquez & Sancho-Rodríguez 2020). Ces sous-disciplines « des » linguistiques de corpus connaissent toutefois des particularités épistémologiques (voire herméneutiques) qui s'infusent dans les méthodologies.

6 Si l'on se limite à la question de la collecte de données à partir du Web, on peut s'attendre à ce que les défis rencontrés par l'AD et le TAL soient sensiblement les mêmes. Deux de ces défis majeurs, le nettoyage du contenu textuel indésirable (le *boilerplate*) et l'existence de doublons problématiques, sont au cœur du travail de Pomikálek (2011) qui leur consacre sa thèse et développe l'algorithme JusText pour répondre à cette double difficulté. Mais les orientations et les impératifs du TAL et de l'AD diffèrent, ce qui peut mener les chercheurs de ces deux domaines à accepter des contraintes différentes, voire contraires.

7 On imagine que pour les chercheurs en TAL qui travaillent sur des corpus dépassant volontiers le milliard de mots (par exemple, Pomikálek *et al.* 2012, Schäfer & Bildhauer 2012), sacrifier l'exactitude au profit du temps de traitement soit une décision acceptable. Ces deux paramètres sont d'ailleurs des sujets discutés dans la littérature : le premier, l'exactitude du contenu extrait (c'est-à-dire la conservation du contenu textuel pertinent tout en excluant le *boilerplate*), est au cœur des démarches de comparaison de performance des outils (voir par exemple Barbaresi & Lejeune 2020), tandis que le second, relatif au temps de traitement, justifie la recherche de procédés toujours plus rapides (voir par exemple Uzun 2020). Dans cette perspective, concéder dans son corpus la présence de textes dont l'extraction n'aura pas été optimale (des textes qui contiennent un peu de *boilerplate* ou auraient négligé certaines parties pertinentes) sera d'autant plus facile que le nombre d'inexactitudes pourra être dilué dans la masse du nombre de mots et que la portée interprétative de l'entreprise reste relativement faible : l'herméneutique du TAL n'est, à tout le moins, pas égale à celle de l'AD pour qui la question est centrale (Sitri & Barats 2017a : 11). Ceci n'empêche évidemment pas que de nouveaux outils à l'ambition d'être toujours plus performants continuent d'être développés et que le TAL continue de prospérer.

8 Du côté de l'AD en revanche, qu'elle soit outillée ou non, l'attitude du chercheur est *a priori* tendanciellement inverse : la nature même du corpus, le cas échéant, est déterminée à l'avance (par exemple : la production textuelle d'une multinationale pétrolière), ou, si elle est soumise à une construction aléatoire, elle reste thématique (par exemple : la communication environnementale de PME sur leur site Web), justement parce que le propos de la discipline n'est pas formellement linguistique. L'épistémologie de l'AD, en particulier dans son approche critique, est idéologiquement située et porte sur des « dysfonctionnements sociaux » (Maingueneau 2014 : 50) ; elle identifie donc nécessairement les problématiques qu'elle souhaite traiter. Aussi « le corpus est[-il] construit en fonction des questions et des hypothèses de recherche » (Sitri & Barats 2017b : 41). Ceci implique que la question de l'exhaustivité devient plus importante, dans la mesure où il ne s'agit pas de quantifier, par exemple, la prévalence de faits de langue par rapport à d'autres, mais *le sens* de cette prévalence en discours, en la mesurant à l'aune d'autres théories des sciences sociales : l'interprétation subséquente est ainsi directement dépendante des données disponibles. À cela s'ajoute le fait que les corpus en AD sont généralement courts, du moins par rapport à l'échelle des corpus de plusieurs milliards de mots du TAL : le corpus de rapports de durabilité d'Aiezza (2015) qui rassemble 258 productions émanant de 67 entreprises approche les 6,5 millions de *tokens* « seulement ». Un tel corpus, déjà considérable, ne représente pourtant qu'un millième des gros corpus de TAL, alors que 6 millions de mots équivalent déjà au double d'une œuvre d'ampleur comme la totalité des *Rougon-Macquart* de Zola (Brunet 1985). Cette brièveté comparative des corpus en AD (qui peut

être encore plus significative en fonction de l'objet étudié) est une conséquence logique de la circonscription du sujet à un ensemble défini : on ne peut pas étendre le corpus *ad libitum* tout en lui permettant de conserver sa cohérence. Il en résulte naturellement que le « coût statistique » d'une absence ou d'un ajout devient mécaniquement plus grand et que sa portée déséquilibrante s'accroisse de la même façon, et il paraît logique de s'attendre à ce que l'analyste soit plus enclin à sacrifier le temps (d'extraction, de (post-)traitement…) au profit de l'exactitude (des données). Ceci est probablement d'autant plus vrai que l'analyse s'opère de façon contrastive (entre plusieurs corpus ou sous-corpus), où chaque valeur doit être comprise de façon relative et non plus absolue. De fait, le commentaire de Sinclair (1991 : 18), selon qui « *a corpus should be as large as possible, and should keep on growing* », n'est pas nécessairement applicable à l'AD qui peut avoir toute légitimité à travailler sur un corpus fermé représentant un état exhaustif de la production textuelle relative à une question (par exemple : l'ensemble des discours d'un candidat pendant sa campagne électorale). Enfin, l'agrandissement perpétuel d'un corpus peut également être dommageable aux méthodes de l'ADT, dont certaines mesures, comme l'analyse factorielle des correspondances, sont directement dépendantes, pour leur précision, du nombre de textes constitutifs du corpus (et de leur longueur).

Revue de la littérature

9 Il semble qu'il y ait assez peu de travaux qui se soient penchés sur l'extraction de contenu textuel à partir du Web dans un objectif d'analyse du discours. Pour Mautner (2005), l'AD, et plus spécifiquement son approche *critique* (dorénavant CDA), était au début du XXI^e^ siècle peu intéressée par la communication électronique, une attitude incohérente avec la portée intrinsèquement sociale de la CDA. Les choses ont cependant évolué, et, dans le domaine de la communication environnementale ou de la responsabilité sociétale des entreprises (RSE) par exemple, de nombreux travaux en AD ou CDA portent, soit sur des supports entièrement dématérialisés tels que la page Web (Pollach 2003, Caimotto & Molino 2011, Fernández-Vázquez & Sancho-Rodríguez 2020), soit sur des documents qui, tout en existant sur le Web, peuvent avoir vocation à être imprimés, et donc diffusés différemment, tels que des rapports RSE ou de développement durable (Jaworska & Nanda 2016, Yu & Bondi 2017, Sun *et al.* 2018). Cependant, à l'exception de Fernández-Vázquez et Sancho-Rodríguez (2020), qui précisent avoir recours à SketchEngine pour l'extraction de données et la suppression de doublons (bien que le programme échoue, de leur propre aveu, à correctement télécharger l'ensemble du contenu des pages Web indiquées [*ibid.* : 5]), les travaux ne s'attardent jamais sur les procédures d'extraction et de nettoyage utilisées, pour autant que nous ayons pu en juger.

10 Évidemment, les cas de figure en matière d'extraction sont multiples et dépendent, notamment, de l'identification *a priori* des textes étudiés ainsi que du genre dont ils relèvent ou du support sur lequel ils sont exprimés. Autrement dit, la délimitation des différentes productions textuelles, préalable essentiel à toute extraction, est variable et s'inscrit dans un continuum allant du plus au moins distinct. On retrouvera, à une extrémité du spectre, les discours comme ceux du président de la République en France, dont les transcriptions sont disponibles sur le site de l'Élysée, ou encore les œuvres d'un auteur donné, qui sont accessibles en ligne via des bibliothèques telles que

Wikisource dès lors qu'elles sont tombées dans le domaine public. Dans de tels cas, le contenu paratextuel (Genette 1982), assimilable au *boilerplate* du point de vue de son indésirabilité, se distingue aisément du corps du texte, le contenu textuel pertinent. À l'autre extrémité du spectre se trouvent des productions aux frontières plus floues qui exploitent toute la créativité de l'outil informatique : on pense par exemple à la publication en ligne de rapports de développement durable riches en éléments sémiotiques, soumis à une mise en page particulièrement variée et à la ponctuation irrégulière, ou même à la simple page Web sur l'un ou l'autre site, parée d'un ensemble d'éléments indésirables (bandeaux en en-tête et pied de page, menus latéraux, fenêtres pop-up, fils d'Ariane, illustrations et leurs légendes, vidéos incrustées…). Plus largement, l'accès au texte, avant même les défis posés par sa délimitation sur un support donné, peut être source de difficultés. C'est le cas lorsque l'on s'intéresse à la communication d'une organisation sur le Web, dont les URLs ne sont pas connues à l'avance et doivent être obtenues par listing. Les listes générées ne peuvent guère être toutes visitées lorsque leur nombre devient trop important, ce qui complique leur simple sélection pour ajout au corpus (par exemple, si l'on travaille sur les pages « RSE et développement durable » d'un ou plusieurs site(s) sans connaître l'ensemble de ces pages *a priori*).

11 Du côté du TAL et des sciences informatiques plus généralement, en revanche, il existe plusieurs travaux relatifs aux procédures d'extraction automatisées à partir du Web. Dans le cas du TAL, il s'agit d'une problématique centrale de la discipline (Al-Ghuribi & Alshomrani 2013), mais les procédés connaissent des applications qui ne sont pas seulement linguistiques (Khder 2021). On nomme « *scraping* » la procédure d'extraction automatisée de données à partir du Web (Singrodia *et al.* 2021), pour laquelle il existe plusieurs méthodes : librairies codées dans des langages de programmation dédiés accessibles aux développeurs, infrastructures logicielles (*frameworks*) et *standalones* (solutions prêtes à l'emploi ; *ibid.*).

12 La littérature offre plusieurs typologies en matière de *scraping*. Pusdekar et Chhaware (2014) opèrent leur classification en fonction des degrés d'automatisation, quand Al-Ghuribi et Alshomrani (2013) ou Barbaresi (2021) distinguent entre les modes opératoires fondamentaux des différentes techniques. Le Détricoteur que nous présentons dans la section suivante participe d'une procédure semi-automatisée exploitant la structure en arbre du Document Object Model (*DOM-tree*). Une telle méthode, comme le relèvent Pusdekar et Chhaware (2014 : 191), requiert un important travail manuel préalable qui peut être consommateur de temps, ce qui, comme nous le suggérons par la suite, rend le procédé probablement inadapté aux besoins des linguistes travaillant en TAL, mais pertinent pour les analystes du discours.

13 Le DOM est une interface qui présente et organise un document HTML sous forme d'arbre, selon une hiérarchie des balises qui le composent. Il permet de déterminer les relations qu'entretiennent les différents objets (*nodes*) entre eux (parents, enfants, frères). Le recours aux *DOM-trees* est présenté comme la façon la plus efficace d'identifier les balises HTML avant extraction des données (Tripathy *et al.* 2012 : 3). Plusieurs travaux proposent des algorithmes exploitant le DOM pour l'extraction de données. Nous précisons qu'à l'exception du Gromoteur (Gerdes 2014), de Sketch Engine (Kilgarriff *et al.* 2014) et de la version en ligne de JusText (Pomikálek 2011), nous n'avons pas eu la possibilité d'utiliser ces algorithmes : nos observations reposent sur les informations obtenues à partir des références consultées.

14 La méthode d'extraction développée par Gupta *et al.* (2003) répond à certaines problématiques précises et évite l'extraction de contenu textuel provenant de publicités en mettant régulièrement à jour une liste noire d'annonceurs ; elle reproduit également la mise en page originale du document, hors de tout contenu jugé indésirable (scripts, images, liens…). Cette dernière fonctionnalité, en particulier, n'est pas immédiatement pertinente dans le cas d'un traitement subséquent des données à l'aide de programmes de textométrie – ce pour quoi le Détricoteur a initialement été développé – mais est susceptible de faciliter un contrôle visuel ultérieur pour l'application d'une norme de dépouillement, en particulier pour les ajouts de ponctuation (le respect de la mise en page d'origine facilitant la lecture du contenu extrait). L'attention portée à l'élimination des publicités n'est également pas une fonctionnalité du Détricoteur, dont l'objectif initial était d'extraire le contenu textuel de sites de petites et moyennes entreprises, donc libres de toute publicité. L'algorithme de Nethra *et al.* (2014), quant à lui, exploite l'apprentissage automatique (*machine learning*) pour générer des règles d'extraction à partir du DOM dans le but d'éviter la présence de bruit en sortie. En fin de compte, l'algorithme présenté par Mehta et Narvekar (2015) paraît être celui qui se rapproche le plus du fonctionnement du Détricoteur, dans le sens où il permet également l'exploitation d'une liste d'URLs préalables ainsi que l'exclusion de certains *nodes* pour l'extraction.

15 Toujours sur la base des *DOM-trees*, des *scrapers* reposant sur des procédés d'exploitation de *visual cues* (permettant la sélection, directement, des parties de texte jugées pertinentes sur une page HTML) ont été développés (Hong *et al.* 2010, Tripathy *et al.* 2012). Ces outils sont réputés produire des résultats plus pertinents que ceux reposant sur les seules balises HTML (Hong *et al.* 2010 : 168). Un logiciel, le Gromoteur (Gerdes 2014), utilise un procédé de ce type pour l'extraction de contenu textuel à partir de pages Web. La sélection visuelle des parties de texte pertinentes pour l'usager repose sur une règle d'inclusion unique (c'est la partie de texte retenue par l'utilisateur qui est extraite, mais il est impossible de cumuler les zones sélectionnées) qui s'applique à l'ensemble des parties de texte régies par les mêmes balises HTML dans le reste de la liste d'URLs. Il faut reconnaître à de telles méthodes une facilité d'utilisation et une intuitivité bienvenues, complétées par la possibilité de naviguer visuellement au travers de l'ensemble des URLs de la liste. Toutefois, le fonctionnement par règle d'inclusion unique empêche l'affinage progressif des parties de texte sélectionnées et fait courir le risque de négliger du contenu pertinent qui serait régi par des balises différentes dans d'autres pages (risque d'autant plus grand que la navigation d'un grand nombre d'URLs peut s'avérer fastidieuse). À l'inverse, le fonctionnement par règles d'exclusion cumulables (comme c'est le cas du Détricoteur) signifie que les parties de texte qui seront extraites ne seront que celles qui ne sont pas spécifiquement marquées comme indésirables. Par conséquent, le seul risque pour l'utilisateur est de se retrouver avec du contenu indésirable, qui toutefois pourra être retiré par la suite en réimportant et étoffant la liste de règles d'exclusion. Comme le soulignent Mehta et Narvekar (2015 : 1), « *If Web pages were written according to a common template, it could easily extract content simply by writing a regular expression* », un vœu pieu dont ils précisent immédiatement après le caractère illusoire. Cette irrégularité architecturale dénoncée par les auteurs ne s'exprime d'ailleurs pas seulement d'un nom de domaine à l'autre, mais parfois aussi entre les URLs d'un même nom de domaine. Le Détricoteur répond ainsi mieux aux problèmes provoqués par de telles irrégularités, puisqu'il

permet une avancée progressive dans les textes. Ce procédé implique toutefois un fonctionnement par essai-erreur qui peut être laborieux.

16 Plusieurs autres algorithmes ont été développés, éventuellement dans le cadre explicite du traitement automatique du langage. Certains ont fait l'objet d'une comparaison formelle de leurs performances (Barbaresi & Lejeune 2020) : c'est notamment le cas de Boilerpy3 (Kohlschütter *et al.* 2010), Dragnet (Peters & Lecocq 2013), JusText (Pomikálek 2011) ou encore News-Please (Hamborg *et al.* 2017). À ces outils s'en ajoutent encore de nouveaux, tels qu'UzunText (Uzun 2020) ou Trafilatura (Barbaresi 2021). Ces objets représentent en fait des « *packages* », c'est-à-dire des modules de code emportant une fonctionnalité définie, tous développés en Python (à l'exception d'UzunText) – ce qui par ailleurs tend à confirmer les observations de Khder (2021) sur la prévalence de ce langage en matière de *Web scraping*. Ce ne sont donc pas des logiciels « clés en main » qui seraient directement utilisables par un analyste profane, non formé aux spécificités du développement informatique, mais bien des solutions dédiées à des utilisateurs dotés d'un savoir-faire technique spécifique. Pour cette raison, les différents algorithmes n'ont pas fait l'objet d'essai dans le cadre de cet article, attendu que son objectif n'est pas de se prononcer sur leur performance, mais bien de présenter un outil utilisable par les linguistes non-informaticiens[1]. Relevons toutefois, à la suite de Lejeune et Barbaresi (2020 : 47), que « le fait qu'il y ait autant d'outils disponibles est en réalité un indicateur de la disparité dans la qualité des résultats obtenus » : aucun programme ne paraît pouvoir se prévaloir d'une efficacité maximale, de sorte que le recours à un logiciel moins automatisé tel que le Détricoteur, au-delà de l'argument de facilité d'utilisation, soit justifié. Mentionnons enfin la plateforme en ligne Sketch Engine (Kilgarriff *et al.* 2014) qui pour sa fonctionnalité d'extraction du contenu textuel d'URLs intègre le module JusText pour la suppression du contenu indésirable (le *boilerplate*), mais dont les performances ont été critiquées par Fernández-Vázquez et Sancho-Rodríguez (2020). Outre les commentaires de ces chercheurs, on peut également regretter, avec cette méthode, un manque de contrôle sur le processus d'extraction.

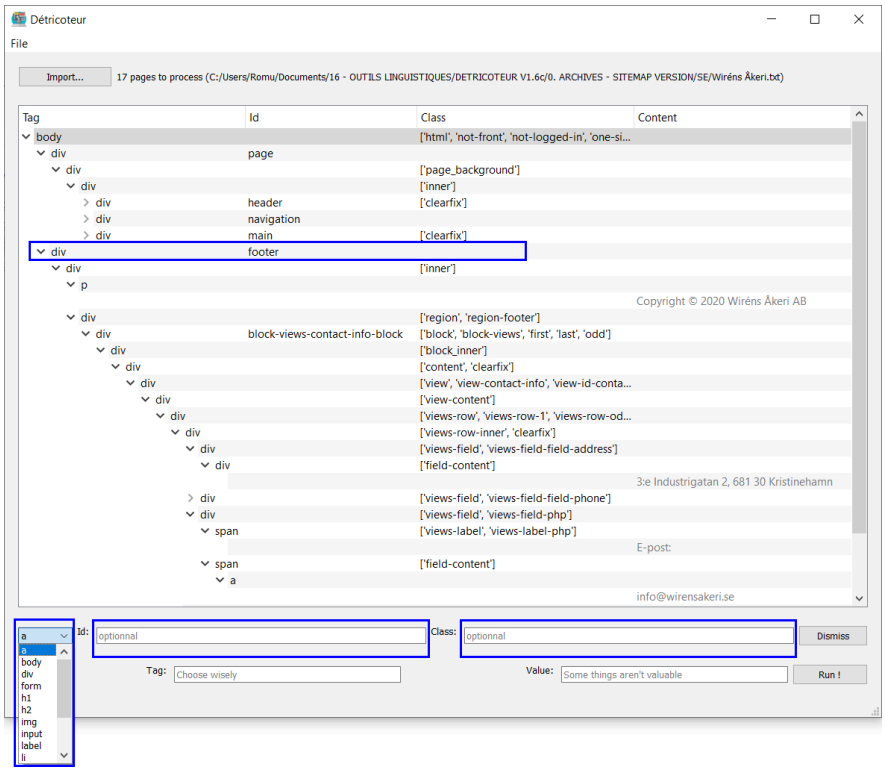
Le Détricoteur : un outil d'extraction de contenu textuel

17 Le Détricoteur est un logiciel d'analyse et d'extraction textuelle de pages HTML. Il présente une unique interface qui regroupe ses principales fonctionnalités :

- Sélection d'un corpus de pages HTML
- Affichage de la structure d'une page HTML (*DOM-tree*)
- Ajout de règles d'exclusion
- Exécution du processus d'extraction

18 L'utilisation du Détricoteur ne nécessite pas une connaissance approfondie du HTML, ni de quelque langage informatique que ce soit. La compréhension du DOM et de la structure des pages HTML est néanmoins conseillée, bien que l'organisation « en cascade » des pages HTML, que l'affichage dans la fenêtre du Détricoteur reproduit, rende cette compréhension intuitive. Le DOM est une norme établie par le World Wide Web Consortium (W3C) qui permet de représenter un document (ici HTML) sous la forme d'un arbre hiérarchique (figure 1). Ainsi représentés, les documents sont plus faciles à parcourir et analyser pour des scripts tels que celui du Détricoteur.

Figure 1. Capture d'écran du Détricoteur



19 Au lancement du Détricoteur, l'utilisateur fournit d'abord une liste d'URLs partageant la même base et qui constitue le corpus à analyser. Le programme parcourt la structure de la première page de la liste fournie et l'affiche sous la forme d'un arbre hiérarchique dont chaque nœud est une balise HTML. Lorsque l'information existe, le programme affiche également pour chaque balise la valeur des propriétés *id*, *class* et *content*. Ces propriétés permettent généralement aux navigateurs Web d'appliquer des règles d'affichage et de comportement.

20 L'utilisateur peut ensuite ajouter des règles d'exclusion. Ces règles permettront d'omettre le contenu des balises qui les régissent lors de l'extraction. Une règle d'exclusion prend la forme d'un ensemble de valeurs (*tag*, *id* et *class*). La seule valeur à indiquer obligatoirement est la nature de la balise (*tag*), que l'utilisateur choisit parmi une liste déroulante. Il est possible de préciser la règle en ajoutant des valeurs pour les propriétés d'*id* et de *class* lorsqu'elles existent, ce qui permet en outre d'affiner l'exclusion, étant entendu que le programme ne distinguera pas entre les différentes valeurs d'*id* et *class* d'une même balise *tag* si aucune précision n'est apportée en ce sens. Ainsi, pour deux ensembles régis par une balise et, respectivement, deux *id* 'header' et 'footer', la règle d'exclusion à deux valeurs « div, footer » n'exclura que le contenu textuel régi par la balise du deuxième ensemble, tandis que la règle à une seule valeur « div » exclura tous les contenus régis par une balise indépendamment des propriétés suivantes, cf. figure 1. La valeur *class* participe du même fonctionnement que la valeur *id*, et le renseignement des trois valeurs (sous réserve qu'elles existent – la balise *tag* étant la seule à être indispensable) permet d'affiner encore la règle d'exclusion.

21 Le Détricoteur applique ensuite les règles éditées à l'ensemble des pages du corpus de l'utilisateur. Par souci de performance, le traitement des pages est parallélisé. Une fois

chaque page analysée, le traitement agrège les données des traitements unitaires, à des fins statistiques et d'information notamment. À la suite de l'extraction, un dossier est généré ; il contient une copie HTML des pages extraites dans un sous-dossier (permettant leur réextraction ultérieure, même si le site d'origine devait changer), l'exportation au format .txt du contenu textuel extrait, respectant l'ordre d'apparition des URLs de la liste d'origine, le fichier des règles d'extraction (qui peut être réimporté préalablement à une nouvelle extraction par souci de gain de temps) et une liste du nombre de « mots » pour chacune des URLs. Les URLs extraites sont automatiquement adossées d'un balisage compatible avec une lecture par Lexico ou Le Trameur (selon une structure attribut-valeur <X=Y-#>, où « X » et « Y » sont des paramètres réglables par l'utilisateur, et « # » un numéro adjoint automatiquement par le programme). Comme il n'existe pas de norme unique pour l'architecture HTML des sites Internet, il est déconseillé d'appliquer le Détricoteur à des listes contenant plusieurs noms de domaine différents : la procédure, bien que techniquement possible, ne pourra pas donner de résultats viables.

22 Parce qu'il applique systématiquement à toutes les URLs de la liste les règles établies d'après l'observation d'une page unique, ce fonctionnement est particulièrement efficace sur un corpus dont les différents documents présentent la même structure. C'est le cas sur les sites de type blog et/ou créés grâce à un outil d'aide à la gestion de contenu (*Content Management System*, CMS). Ces outils sont fréquemment utilisés pour les sites institutionnels, notamment lorsque la création de contenu sur le Web n'est pas le cœur de métier de la société. Au contraire, sur des sites à la présentation changeant d'une page à l'autre, le recours au Détricoteur ne permettra pas une bonne analyse. Depuis l'avènement du Web 2.0, les pages Web contiennent de plus en plus de contenu dynamique, qui peut rendre plus difficile la lecture du DOM d'un document en multipliant le nombre de balises et leurs imbrications.

Conclusion : forces et faiblesses du Détricoteur

23 Le Détricoteur n'a pas la prétention de répondre à tous les besoins en matière d'extraction de contenu textuel : il s'agit d'un outil mieux adapté à un public et une utilisation spécifiques, qui dispose de forces et de faiblesses qui lui sont propres.

24 Sa principale force est son adéquation avec les besoins d'utilisateurs non-développeurs, voire non-spécialistes en informatique. En tant que solution logicielle prête à l'emploi, il est directement accessible aux usagers qui ne maîtrisent pas de langage informatique. L'interface a été conçue pour être aussi simple et intuitive que possible, de sorte que la prise en main du logiciel est immédiate.

25 En travaillant à partir d'une liste d'URLs découlant d'un nom de domaine initial (liste que l'on peut obtenir à l'aide de programmes dédiés facilement accessibles sur Internet), il n'est pas nécessaire pour l'utilisateur de visiter manuellement toutes les pages d'un nom de domaine donné : ainsi, certaines pages éventuellement pertinentes mais difficiles d'accès sur le Web deviennent aussi accessibles que les autres, un obstacle qu'une méthode par copié/collé ne permettrait pas d'éviter, puisqu'on ne copie-colle par définition que ce que l'on visite. Ceci permet, par exemple, d'effectuer une recherche par mots-clés sur l'ensemble du contenu textuel pertinent une fois le paramétrage des règles opéré et l'extraction effectuée, et de ne sélectionner que les URLs pertinentes pour l'analyse.

26 La possibilité de réimporter le fichier de règles permet un affinage progressif du travail d'exclusion si certaines URLs continuent de présenter du contenu indésirable : il suffit alors de replacer en tête de liste l'URL « fautive » et de déterminer la ou les balise(s) qui doivent à leur tour être exclues.

27 Surtout, par rapport à une extraction des données manuelle par copié-collé, le Détricoteur permet de mieux circonscrire le texte désirable sur la base de critères plus objectifs (les balises). Ceci, combiné au fait que le programme sauvegarde automatiquement les fichiers au format HTML, permet la reproductibilité de la recherche par d'autres utilisateurs (sous réserve que les fichiers de règles soient communiqués), une question plusieurs fois soulevée lorsqu'il est question de données issues du Web (par exemple, Mautner 2005 : 818, Lüdeling *et al.* 2007 : 11).

28 Le Détricoteur n'est toutefois pas adapté à tous les usages : il est principalement destiné aux analystes du discours, probablement plus enclins à sacrifier le temps de traitement au profit de l'exhaustivité des données que l'inverse : parce qu'il fonctionne selon une méthode essai-erreur qui peut se révéler fastidieuse lors des architectures particulièrement complexes (des extractions menées sur plus de 200 noms de domaine différents suggèrent toutefois que de tels cas de figure sont plutôt rares, même s'ils existent), et parce qu'un nettoyage manuel complémentaire peut se révéler nécessaire (par exemple pour éliminer du *boilerplate* qui aurait pu être subordonné à des balises régissant du contenu textuel pertinent sur d'autres URLs), il est plutôt adapté aux petits corpus, qui ne cherchent pas à rassembler plus de quelques dizaines de noms de domaine au maximum – certainement pas, en tout cas, aux corpus gigantesques qui visent le milliard de mots ou plus. Plutôt qu'un logiciel d'extraction automatisée, il s'agit en fait d'une procédure d'extraction semi-automatisée, qui nécessite un paramétrage initial à adapter à chaque nouveau nom de domaine.

29 Enfin, l'usage du Détricoteur est avant tout destiné aux analystes du discours ou aux praticiens de l'ADT qui ont au préalable circonscrit l'objet de leurs recherches. Deux usages paraissent s'imposer dans l'immédiat : soit la constitution d'un corpus d'URLs « thématiques » rassemblant plusieurs noms de domaine (nécessitant donc autant d'extractions) et permettant une analyse du traitement de la thématique en question (Dalodiere 2023), soit la constitution d'un corpus à partir d'un nom de domaine unique pour étudier, par exemple, la cohérence de la communication d'une seule organisation sur l'ensemble de son site Internet. Dans un cas comme dans l'autre, il est nécessaire de garder en tête que le Détricoteur n'opèrera d'extraction qu'à partir de la liste d'URLs qui lui aura été fournie initialement, ce qui implique d'obtenir une telle liste dans un premier temps.

BIBLIOGRAPHY

Aiezza M. C. (2015). « "We May Face the Risks" … "Risks that Could Adversely Affect our Face." A Corpus-Assisted Discourse Analysis of Modality Markers in CSR Reports », *Studies in Communication Sciences* 15(1) : 68-76.

Alexander R. J. (2009). *Framing Discourse on the Environment : A Critical Discourse Approach*. New York / Abingdon : Routledge.

Al-ghuribi S. M. & Alshomrani S. (2013). « A Comprehensive Survey on Web Content Extraction Algorithms and Techniques », *2013 International Conference on Information Science and Applications (ICISA)*, Changsha, Chine, 8-9 nov. 2013 : 1-5.

Barbaresi A. & Lejeune G. (2020). *Out-of-the-Box and Into the Ditch ? Multilingual Evaluation of Generic Text Extraction Tools.* Proceedings of the 12th Web as Corpus Workshop, Language Resources and Evaluation Conference (LREC 2020), Marseille, France, 11-16 mai 2020 : 5-13.

Barbaresi A. (2021). *Trafilatura : A Web Scraping Library and Command-Line Tool for Text Discovery and Extraction.* Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing : System Demonstrations [en ligne], 1-6 août 2021 : 122-131.

Bernard T. (2015). *A critical analysis of corporate reports that articulate corporate social responsibility*. Thèse de doctorat, Stellenbosch University, Stellenbosch.

Brunet É. (1985). « On a compté trois millions de mots chez Zola. Et alors ? », *Computers in literary and linguistic Computing*, Champion Slatkine, 63-91.

Caimotto M. C. & Molino A. (2011). « Anglicisms in Italian as Alerts to Greenwashing : A Case Study », *Critical Approaches to Discourse Analysis across Disciplines* 5(1) : 1-16.

Dalodiere R. (2023). *Analyse du discours environnemental et sociétal de PME scandinaves et francophones : une approche textométrique*. Thèse de doctorat, Université de Mons, Mons.

De Schryver G. D. (2002). « Web for/as corpus : a perspective for the African languages », *Nordic Journal of African Studies* 11(2) : 266-282.

Fernández-Vázquez J. & Sancho-Rodríguez Á. (2020). « Critical discourse analysis of climate change in IBEX 35 companies », *Technological Forecasting and Social Change* 157, article 120063.

Gatto M. (2011). « The 'body' and the 'Web'. The Web as corpus ten years on », *ICAME Journal* 35 : 35-58.

Genette G. (1982). *Palimpsestes.* Paris : Seuil.

Gerdes K. (2014). *Corpus collection and analysis for the linguistic layman : The Gromoteur.* JADT 2014 : 12e Journées internationales d'Analyse statistique des Données Textuelles, Paris, France, 3-6 juin 2014 : 261-269.

Gupta S., Kaiser G. E., Neistadt D. & Grimm P. (2003). *DOM-based content extraction of HTML documents*. Proceedings of the 12th international conference on World Wide Web (WWW '03), Budapest, Hongrie, 20-24 mai 2003 : 207-214.

Habert B., Nazarenko A. & Salem A. (1997). *Les linguistiques de corpus*. Paris : Armand Colin.

Hamborg F., Meuschke N., Breitinger C. & Gipp B. (2017). « News-please : A generic news crawler and extractor », in Gaede M. *et al.* (éd.) *Proceedings of the 15th International Symposium of Information Science*, Berlin, Allemagne, 13-15 mars 2017 : 218-223.

Hong J. L., Siew E. & Egerton S. (2010). *ViWER- data extraction for search engine results pages using visual cue and DOM Tree*. 2010 International Conference on Information Retrieval & Knowledge Management (CAMP), 17-18 mars 2010, Shah Alam, Malaisie : 167-172.

Jaworska S. & Nanda A. (2016). « Doing well by talking good ? A topic modelling-assisted discourse study of corporate social responsibility », *Applied Linguistics* 39(3) : 373-399.

Khder M.A. (2021). « Web Scraping or Web Crawling : State of Art, Techniques, Approaches and Application », *International Journal of Advances in Soft Computing and its Application* 13(3) : 144-168.

Kilgarriff A. & Grefenstette G. (2003). « Introduction to the Special Issue on the Web as Corpus », *Computational Linguistics* 29(3) : 333-347.

Kilgarriff A., Baisa V., Busta J., Jakubíček M., Kovář V., Michelfeit J., Rychlý P. & Suchomel V. (2014). « The Sketch Engine : ten years on », *Lexicography* 1 : 7-36.

Kohlschütter C., Fankhauser P. & Nejdl W. (2010). *Boilerplate detection using shallow text features*. Proceedings of the 3rd ACM International Conference on Web Search and Data Mining, New York, États-Unis, 4-6 fév. 201 : 441-450.

Lejeune G. & Barbaresi A. (2020). *Bien choisir son outil d'extraction de contenu à partir du Web*. Actes de la 6e conférence conjointe Journées d'Études sur la Parole (JEP, 31e édition), Traitement Automatique des Langues Naturelles (TALN, 27e édition), Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RÉCITAL, 22e édition), Nancy, France, 8-19 juin 2020 : 46-49.

Lüdeling A., Evert S. & Baroni M. (2007). « Using Web data for linguistic purposes », *in* Hundt M., Nesselhauf N. & Biewer C. (éd.) *Corpus linguistics and the Web*. Amsterdam / New York : Rodopi, 7-24.

Maingueneau D. (1996). *Les termes clés de l'analyse du discours*. Paris : Seuil.

Maingueneau D. (2014). *Discours et analyse du discours*. Paris : Armand Colin.

Mautner G. (2005). « Time to get wired : Using Web-based corpora in critical discourse analysis », *Discourse & Society* 16(6) : 809-828.

Mehta B. & Narvekar M. (2015). *DOM tree based approach for Web content extraction*. 2015 International Conference on Communication, Information & Computing Technology (ICCICT), Mumbai, Inde, 16-17 jan. 2015 : 1-6.

Nethra K., Anitha J. & Thilagavathi G. (2014). « Web Content Extraction Using Hybrid Approach », *ICTACT Journal on Soft Computing* 4(2) : 692-696.

Peters M. E. & Lecocq D. (2013). *Content extraction using diverse feature sets*. 22nd International World Wide Web Conference, Rio de Janeiro, Brésil, 13-17 mai 2013 : 89-90.

Pollach I. (2003). « Communicating Corporate Ethics on the World Wide Web - A Discourse Analysis of Selected Company Web Sites », *Business & Society* 42(2) : 277-287.

Pomikálek J. (2011). *Removing Boilerplate and Duplicate Content from Web Corpora*. Thèse de doctorat, Masaryk University, Brno.

Pomikálek J., Jakubíček M. & Rychlý P. (2012). *Building a 70 billion word corpus of English from ClueWeb*. 8th International Conference on Language Resources and Evaluation, Istanbul, Turquie, 21-27 mai 2012 : 502-506.

Pusdekar S. & Chhaware S. P. (2014). *Using Visual Clues Concept for Extracting Main Data from Deep Web Pages*. 2014 International Conference on Electronic Systems, Signal Processing and Computing Technologies, Nagpur, Inde, 9-11 jan. 2014 : 190-193.

Schäfer R. & Bildhauer F. (2012). *Building large corpora from the Web using a new efficient tool chain*. 8th International Conference on Language Resources and Evaluation, Istanbul, Turquie, 21-27 mai 2012 : 486-493.

Sinclair J. (1991). *Corpus, Concordance, Collocation*. Oxford : Oxford University Press.

Sinclair J. (2004). « Corpus and Text — Basic Principles », *in* Wynne M. (éd.) *Developing Linguistic Corpora : A Guide to Good Practice*. Oxford : Oxbow Books, 1-16.

Singrodia V., Mitra A. & Paul S. (2019). *A Review on Web Scrapping and its Applications*. 2019 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, Inde, 23-25 janvier 2019 : 1-6.

Sitri F. & Barats C. (2017a). « Introduction », *in* Née É. (dir.) *Méthodes et outils informatiques pour l'analyse des discours*. Rennes : PUR, 9-16.

Sitri F. & Barats C. (2017b). « Constituer un corpus en analyse du discours, un moment crucial », *in* Née É. (dir.) *Méthodes et outils informatiques pour l'analyse des discours*. Rennes : PUR, 41-62.

Sun Y., Jin G., Yang Y. & Zhao J. (2018). « Metaphor Use in Chinese and American CSR Reports », *IEEE Transactions on Professional Communication* 61(3), article 8361923 : 295-310.

Tanguy L. (2013). « La ruée linguistique vers le Web », *Texto ! Textes et Cultures* [en ligne] XVIII(4).

Tripathy A. K., Joshi N., Thomas S., Shetty S. & Thomas N. H. (2012). *VEDD- a visual wrapper for extraction of data using DOM tree*. 2012 International Conference on Communication, Information & Computing Technology (ICCICT), Mumbai, Inde, 19-20 oct. 2012 : 1-6.

Uzun E. (2020). « A Novel Web Scraping Approach Using the Additional Information Obtained From Web Pages », *IEEE Access* 8 : 61726-61740.

Yu D. & Bondi M. (2017). « The Generic Structure of CSR Reports in Italian, Chinese, and English : A Corpus-Based Analysis », *IEEE Transactions on Professional Communication* 60(3) : 273-291.

NOTES

1. La seule exception étant, éventuellement, la version en ligne de JusText (accessible ici : https://nlp.fi.muni.cz/projects/justext/ ; consultée le 6 octobre 2024), qui ne permet toutefois d'ouvrir qu'une seule URL à la fois, et dont les essais ont montré qu'elle échouait à extraire certains éléments jugés pertinents (notamment, des éléments de listes à puces) tout en incluant d'autres parties jugées non pertinentes (comme les bandeaux relatifs au dépôt de cookies).

ABSTRACTS

There are a number of tools for extracting textual content on the Internet. Many of these tools were designed by researchers in the field of natural language processing, and are available as "packages", which are sets of code files that developers have no trouble using but that may prove out of reach for laymen. Independent software solutions are few, and unlikely to satisfy the needs of researchers in the field of discourse analysis. In this article, we introduce a semi-automated, textual data extraction software, Le Détricoteur, which aims to meet the epistemological requirements of discourse analysis while also being accessible to non-specialist users who are not familiar with coding.

Il existe aujourd'hui de nombreux outils en matière d'extraction du contenu textuel sur Internet. Beaucoup de ceux-ci ont été conçus à l'initiative de chercheurs travaillant en traitement automatique du langage, et prennent la forme de « *packages* » : des modules de codes simples à utiliser pour les usagers développeurs, mais inaccessibles aux profanes. Les solutions logicielles indépendantes sont peu nombreuses et ne sont pas susceptibles de répondre aux besoins des chercheurs en analyse du discours. Dans cet article, nous présentons le Détricoteur, un programme d'extraction textuelle semi-automatique, qui cherche à répondre aux contraintes épistémologiques de l'analyse du discours tout en étant utilisable même sans connaissances en matière de code informatique.

INDEX

Keywords: discourse analysis, text extraction, software
Mots-clés: analyse du discours, extraction de données textuelles, logiciel

AUTHORS

ROMUALD DALODIERE

Université de Mons, Service NORD

MANUEL JORDAN

Ingénieur IMAC