Étude du flux de dépendance dans 70 corpus (50 langues) de UD

CHUNXIAO YAN
Université de Paris 3

Mémoire de master 2
Mention : Traitement automatique des langues
Spécialité : Recherche & Développement
Année universitaire 2016-2017

Mémoire dirigé par SYLVAIN KAHANE

Table des matières

Chapitre 1 Introduction

Ce mémoire étudie les propriétés du flux de dépendance[1] dans les 70 treebanks en 50 langues distribués par le projet Universal Dependencies (Nivre et al. 2016). Notre objectif est de mettre en évidence les contraintes mémorielles universelles qui pèsent sur la complexité des structures syntaxiques. Pour cela, nous étudierons dans ces 50 langues les propriétés du flux de dépendance. Les limites, notamment sur le nombre de niveaux d'auto-enchâssement (ang. *center-embedding*) que nous avons trouvées pourraient être causées par la limite de la mémoire immédiate, qui est supposée être de l'ordre de 7±2 éléments selon l'étude psychologique de Miller (1960) .

Tout d'abord, nous allons présenter des travaux permettant de mesurer la complexité des structures syntaxiques à partir de l'analyse en constituance et de l'analyse en dépendance. Puis, nous proposerons des mesures sur les propriétés du flux de dépendance et nos hypothèses. La présentation du projet Universal Dependencies nous permettra ensuite d'avoir une vision basique sur les 70 corpus et leur schéma d'annotation. Nous présenterons aussi notre expérience et les traitements techniques effectués sur les treebanks. Après l'expérience, nous passerons aux résultats et aux discussions en montrant des exemples dans les treebanks chinois, anglais et français, et les perspectives seront proposées à la fin.

Chapitre 2 État de l'art

2.1 Compétence et performance

Notre recherche concerne la performance du langage. D'après l'hypothèse générativiste proposée par Chomsky (1965), la compétence linguistique est commune à tous les locuteurs, elle permet de traiter un ensemble de règles pour produire et comprendre une phrase. L'ensemble des phrases que nous pouvons produire à partir des règles est infini en théorie.

[1] Définition : Le flux de dépendance en une position donnée (entre deux mots d'une phrase) est l'ensemble des dépendances qui relient un mot à gauche de cette position à un mot à droite. (Kahane 2001)

Mais dans la réalité, il peut être contraint par la performance du langage, comme la limite de la capacité cognitive : la mémoire immédiate.

2.2 Grammaire de constituance et profondeur

2.2.1 Grammaire de constituance

La grammaire de constituance est née des travaux de Bloomfield (1933) où il propose l'analyse en constituants immédiats. Dans les années 1950, en partant de la théorie de Bloomfield, Noam Chomsky développe dans *Syntactic structures* (1957) la grammaire générative. Cette approche a fortement influencé le domaine de la grammaire formelle.

Selon Chomsky (1957), la grammaire d'une langue est alors considérée comme l'ensemble des règles qui régissent les constituants à l'intérieur d'une phrase, les règles se différencient entre les différentes langues.

2.2.2 Profondeur (angl. *depth*)

L'une des premières recherches qui cherche à modéliser la langue et qui tient compte de la mémoire à court terme sont celle de Yngve (1960), où est proposé un modèle contenant deux parties, la grammaire qui pourrait être spécifique pour chaque langue, et un mécanisme général permettant de fonctionner avec la grammaire de la langue.

En suivant la conception générativiste, chaque langue possède un ensemble de règles finies, représentable sous la forme d'une grammaire de constituants. Un mécanisme associé avec la mémoire immédiate traite ces règles de grammaire pour former une phrase grammaticale.

Concernant le mécanisme, Yngve (1960) a posé trois questions et a essayé d'y répondre:

"Under what conditions will a finite constituent-structure device fail to operate properly because it has used up its temporary storage capacity ? Is it possible to have a well-behaved grammar, that is , a grammar so restricted that all the sentences generated can actually be produced by a finite constituent structure device with a given temporary memory capacity ? And is it possible that the grammar of English , unlike grammar of algebra, is well behaved? In order to answer these questions we are led to investigate the relation between output sequences and the amount of temporary storage needed to produce them . "

La quantité du stockage temporaire peut être calculée par la profondeur (angl. *depth*) à partir d'un arbre de constituent. Tout d'abord, il faut numéroter à chaque niveau toutes les branches de droite à gauche, à partir de 0 jusqu'à *n*-1. Puis, pour avoir la profondeur *d* de chaque noeud terminal, il faut calculer la somme de toutes les branches menant à ce noeud terminal. La quantité maximale du stockage temporaire pour construire cet arbre est la profondeur de la phrase : $D = d_{max}$ qui est la plus grande valeur de *d* dans cet arbre.

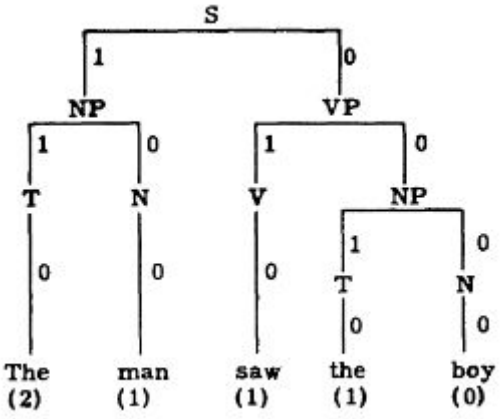


Figure 2-1 Yngve (1960)

Par exemple, dans la figure 2-1, la valeur de *d* du mot "The" est la somme de la branche NP--->T (1) et S--->NP (1), donc 2. Cette valeur indique également que dans l'état de "The", deux éléments (le N et le VP) sont dans la mémoire immédiate, ce qui correspond à la taille de la pile dans la procédure du parsing (voir Nivre et al., 2008). Après avoir calculé toutes les valeurs de *d,* on constate que la plus grande valeur de *d* est de 2, et donc la profondeur de la phrase *D* est de 2.

Cette mesure est basée sur une analyse en grammaire de constituants. L'ensemble des règles de constituance sont contenues dans la mémoire permanente, et les phrases bien formées par des règles devraient avoir une profondeur qui ne dépasse pas une certaine valeur proche ou égale à la limite de la mémoire immédiate. Selon les études de Miller 1956, cette limite est de 7±2 éléments (Yngve 1960).

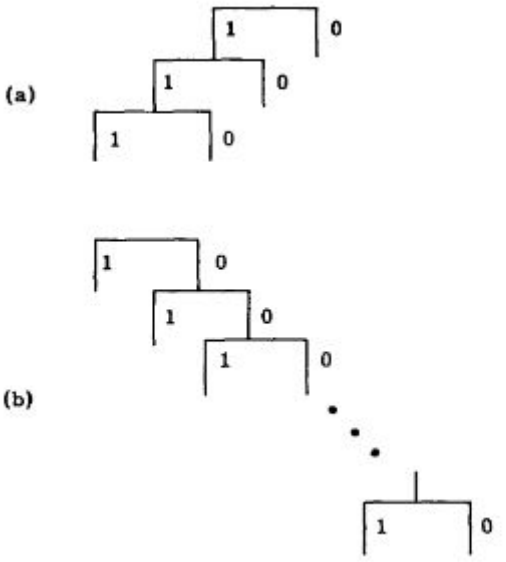


Figure 2-2 Yngve (1960)

La figure 2-2 (Yngve 1960) illustre deux constructions typiques qui peuvent produire des phrases de longueur infinie, tout en respectant la bonne formation grammaticale. La figure 2-2 nous montre ces deux constructions en grammaire de constituants, (a) pour la *construction régressive* qui représente une structure branchant à gauche (angl. *left-branching*) et (b) pour la *construction progressive* qui représente une structure branchant à droite (angl. *right-branching*). Concernant la construction régressive, l'augmentation du niveau d'arbre pourra provoquer une profondeur qui dépasserait la limite maximale. Quant à la construction progressive (b), la valeur des mots à gauche ne change pas lorsque le nombre du niveau de l'arbre est incrémenté : chaque mot serait de valeur 1, hormis le dernier mot ayant pour valeur 0, par conséquent, cette construction pourrait s'étendre indéfiniment en respectant la valeur maximale de profondeur.

L'auteur propose des contraintes pour limiter le nombre des niveaux pour la construction régressive. Il suggère aussi que le modèle pourrait utiliser la représentation progressive pour diminuer la profondeur de certaines constructions spécifiques.

La profondeur sert à tenir compte de la limite mémorielle dans la modélisation, mais cette mesure donne également la possibilité d'avoir un moyen de mesurer ou de confirmer cette limite 7±2, fournie par des recherches en psychologie.

Murata et al. (2001) a suivi cette méthode de profondeur pour analyser les corpus Penn Treebank et SUSANNE. Les branches d'arbre du corpus ne sont pas binaires comme ce qui a

été proposé dans le modèle de Yngve, par conséquent les résultats de la profondeur ne se limitent pas à 7±2 . L'auteur a appliqué deux variantes afin de bien adapter l'analyse de ces deux corpus. L'une est la méthode de Sampson qui considère que toutes les branches d'un noeud n'ont que deux parties : si nous avons les branches: A,B,C,D,E de gauche à droite sous un noeud (voir la figure 2-3), nous prenons A,B,C,D comme un ensemble ayant la valeur 1, et E ayant pour valeur 0, ceci permet de respecter le principe de numérotation en binaire.

Figure 2-3 Murata et al. (2001) méthode de Sampson

L'autre méthode de Murata et al. (2001), distincte de celle de Yngve et de Sampson, est de calculer la somme des valeurs de la branche à partir de S jusqu'au niveau de SN (syntagme nominal), en considérant que le SN est une seule unité cognitive indivisible.

Chacune de ces nouvelles variantes pour le calcul de la profondeur permet d'avoir des résultats qui respectent la limite de 7±2 :

La méthode de Sampson: la profondeur de mot est de 5 au maximum pour le corpus SUSANNE anglais (130,300 mots), et la profondeur de mot et de phrase sont de 7 au maximum pour le Penn Treebank (Marcus et al., 1993) .

La méthode de Murata et al. (2001): pour le Penn Treebank (Marcus et al., 1993) , la profondeur de mot et de phrase est de 10 au maximum, seulement 0.01% des mots ont une profondeur égale ou supérieure à 9.

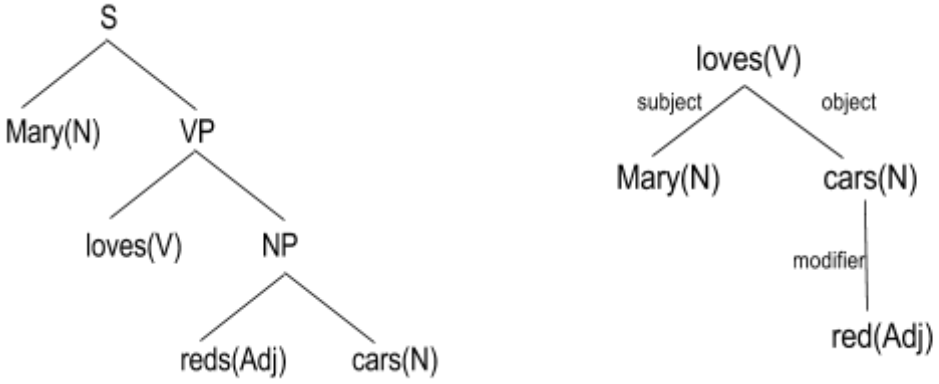
2.3 Syntaxe de dépendance

La syntaxe de dépendance a été fondée par le linguiste français Lucien Tesnière (1893-1954) dans son ouvrage publié à titre posthume *Les éléments de syntaxe structurale* (1959). La notion de dépendance était expliquée dans son ouvrage : "*Les connexions structurales établissement entre les mots des rapports de dépendance. Chaque connexion unit en principe un terme supérieur à un term inférieur. Le terme supérieur reçoit le nom de régissant. Le terme inférieur reçoit le nom de subordonné. Ainsi dans la phrase* ***Alfred parle***, ***parle*** *est le*

régissant et ***Alfred*** *le subordonné.*" La hiérarchie de connexions est montrée par *le stemma* (aujourd'hui on utilise le terme *arbre de dépendance*).

Hays (1958) met en correspondance les arbres de la syntaxe de dépendance avec le groupement en mots de la grammaire de constituants. La grammaire de dépendance a constitué la base de deux importantes théories linguistiques : la 'Word Grammar', développée par Hudson à partir de 1984, et la Théorie Sens Texte d'Igor Mel'cuk, exposée en 1988 dans l'ouvrage *Dependency syntax : Theory and practice*. Elle a été enrichie des travaux de Sylvain Kahane en France, notamment dans l'étude « Grammaires de dépendance formelles et théorie Sens-Texte » (2001).

En comparaison avec la syntaxe en constituants, la syntaxe de dépendance est plus économique à appliquer dans les technologies du traitement automatique des langues. Par conséquent, elle est devenue le modèle de référence pour l'analyse syntaxique automatique (ang. *parsing*) et dans le développement de treebanks syntaxiques. Ci-dessous la figure 2-4, un arbre syntaxique en constituant, et un arbre syntaxique en dépendance.



arbre de constituance *vs* *arbre de dépendance*

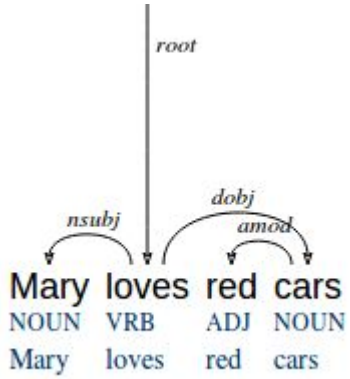
Figure 2-4 *(Kahane & Maziotta, 2015)*

Pour que les données contenues dans l'arbre syntaxique soient traitées informatiquement, celles-ci sont compilées dans des treebanks. Un treebank a cette forme là :

Treebank en syntaxe de dépendance[2] :

numéro	token	lemma	étiquette	gouverneur	relation
1	Mary	Mary	NOUN	2	nsubj
2	loves	love	VRB	0	root
3	red	red	ADJ	4	amod
4	cars	car	NOUN	2	dobj

Il existe aussi la représentation en arbre linéairement ordonné. Arborator, développé par Kim Gerdes (2013), est un outil qui permet de manipuler les treebanks en ligne, avec une représentation en arbre linéairement ordonné. Nous le montrons ci-dessous:



2.4 Complexité linguistique et traitement de la phrase

2.4.1 Les constructions avec auto-enchâssement (ang. *center-embedding*) et la limitation de la mémoire immédiate

La première question concernant la complexité de la phrase et la capacité cognitive est le phénomène de la limitation sur le niveau d'auto-enchâssement dans la phrase. Miller & Chomsky (1963) définissent la construction auto-enchâssée (ang. *center embedding*), comme : « [a] nesting of dependencies, which occurs when a constituent X is embedded in another constituent Y, with material in Y to both the left and right of X. » , et ils rendent compte de l'impossibilité de la compréhension de la phrase en augmentant les niveaux d'auto-enchâssement.

[2] *ici nous ne montrons que les 6 colonnes les plus importantes, mais il existe des treebank standards à 10 colonnes, voire 14. Cela varie en fonction des différents schémas d'annotation.*

Nous montrons ici des exemples en français (Wehrli, E. 1989):

Niveau 0. Le chat miaule.

Niveau 1. Le chat que le chien poursuit miaule.

Niveau 2. Le chat que le chien que l'homme aime poursuit miaule.

D'après ces exemples , nous constatons que l'augmentation des niveaux d'auto-enchâssement complexifie la phrase; Quand nous avons deux niveaux d'auto-enchâssement, la compréhension devient assez difficile.

Richard L.Lewis (1996) a suivi cette idée et recherché la limite d'auto-enchâssement. Il a proposé un modèle d'analyse *NL-Soar*[3] en prenant en compte la vision cross-linguistique, et a montré que "*it is possible to combine general principles of memory with linguistic analyses to produce a detailed model of syntactic working memory that is empirically powerful, and psychologically and functionally grounded*".

D'après les travaux de Lewis, une phrase anglaise a deux niveaux d'auto-enchâssement au maximum. Pour le japonais le nombre total peut atteindre à trois. Cette limitation syntaxique seraient liées aux contraintes de la mémoire immédiate.

2.4.2 SPLT (*the Syntactic Prediction Locality Theory*)

Si nous ne considérons que le nombre de niveaux d'enchâssement (y compris le cas de l'auto-enchâssement) qui pose des difficultés dans le traitement de la phrase, une question inévitable est celle des phrases relatives. En effet, la difficulté de la compréhension d'une extraction d'un sujet et d'une extraction d'un objet ne sont pas les mêmes, selon Gibson (1998) :

"*One well-established complexity phenomenon to be explained by a theory of the relationship between the sentence processing mechanism and the available computational resources is the higher complexity of an object-extracted relative clause compared with a subject-extracted in a Subject-Verb-Object language like English*" ;

"Aphasic stroke patients cannot reliably answer comprehension questions about object-extracted RCs, although they perform well on subject-extracted RCs (Caramazza and zurif, 1976; Caplan and Futter, 1986; Grodzinsky, 1989; Hiekok et al.,1993)"

[3] NL-soar (Lewis, 1993) est un modèle computationnel de compréhension de phrases en temps réel, il est basé sur Soar, la théorie d'architecture cognitive humaine (Newell, 1994;Laird et al. 1987; Rosenbloom et al., 1993)

Exemple de Gibson (1998):

a. extraction d'un objet: *The reporter who the senator attacked admitted the error.*
b. extraction d'un sujet: *The reporter who attacked the senator admitted the error.*

Dans la phrase *a,* le pronom relatif *who* occupe une place d'objet du verbe *attacked*, il s'agit d'une extraction d'un objet; Dans la phrase *b*, le mot *who* occupe une place de sujet du verbe *attacked,* ainsi *b* a une extraction d'un sujet.

Selon Gibson (1998) , deux facteurs influencent la compréhension de la phrase :

Le coût de la mémoire est le nombre de catégories syntaxiques requises pour compléter la chaîne d'entrée actuelle en tant que phrase grammaticale.

Le coût de l'intégration est la distance entre les deux unités linguistiques attachées par une relation de dépendance.

Ces deux notions sont intégrées dans son modèle Syntactic Prediction Locality Theory (SPLT). La différence entre extraction d'un sujet et extraction d'un objet pourrait être expliquée en particulier par la différence du coût d'intégration défini dans le modèle SPLT. Nous reprenons les exemples ci-dessus de Gibson (1998) :

extraction d'un objet : *The reporter who the senator attacked admitted the error.*

-	0	0	0	0	1+2	3	0	0+1

extraction d'un sujet : *The reporter who attacked the senator admitted the error.*

-	0	0	0+1	0	0+1	3	0	0+1

D'après l'analyse du modèle SPLT concernant le coût d'intégration:

Dans la phrase avec extraction d'un objet, le coût d'intégration du mot *attacked* est calculée de façon la ci-dessous :

Il faut calculer la distance de dépendance. Deux relations de dépendances sont touchées :

1. relation sujet liée avec *the senator* : depuis *the senator* jusqu'à *attacked,* il s'agit d'un nouveau référent[4] (*attacked*), la distance est de 1.
2. relation objet liée avec *who* : depuis *who* jusqu'à *attacked ,* il s'agit de deux nouveaux référents (*the senator* et *attacked*), la distance est de 2.

[4] Le verbe est considéré comme référent d'événement, et le syntagme nominal est considéré comme référent de discours.

Ainsi la valeur du coût d'intégration est de 1 (*attacked*) + 2 (*the senator* et *attacked*) .

Si nous regardons le mot *admitted,* nous trouvons une relation sujet liée avec *the reporter*, depuis *the reporter* jusqu'à *admitted*, il s'agit de trois nouveaux référents : *the senator*, *attacked* , et *admitted*. Ainsi la valeur du coût d'intégration est de 3.

Pour la phrase avec extraction du sujet, la valeur du coût d'intégration est: *attacked* a pour coût d'intégration 1, et *admitted* a pour coût d'intégration 3 .

Ainsi, dans la phrase avec extraction d'un objet, le coût d'intégration du mot *attacked* est plus grand que dans la phrase avec extraction d'un sujet, ce qui pourrait complexifier la phrase.

Les concepts présentés dans Gibson (1998) rendent compte des distances de relation de dépendance qui séparent les unités linguistiques d'une phrase. Ces distances de dépendance permettent aussi de mesurer les limites de la mémoire immédiate. Nous en discutons dans la section suivante.

2.4.3 Étude quantitative cross-linguistique

Depuis les travaux de Gibson (1998; 2000), la recherche en linguistique quantitative sur la distance de dépendance a été poursuivie principalement par Haitao Liu. Dans (Liu, 2008) l'auteur a basé sa recherche sur la syntaxe de dépendance.

Selon Liu (2008), "*the linear distance between governor and dependent is defined as 'dependency distance'",* la distance de dépendance est donc calculée à partir du nombre de tokens qui séparent un token de son gouverneur.

Liu (2008) a calculé la distance de dépendance sur un ensemble de treebanks dans 20 langues différentes. La moyenne de la distance de dépendance (MDD) varie selon les langues (et les corpus utilisés) entre 1,798 et 3,662.

La formule de MDD d'une phrase est :

$$MDD(the\ sentence) = \frac{1}{n-1}\sum_{i=1}^{n} |\, DD_i \,|$$

DDi : distance de dépendance entre le token i et son gouverneur

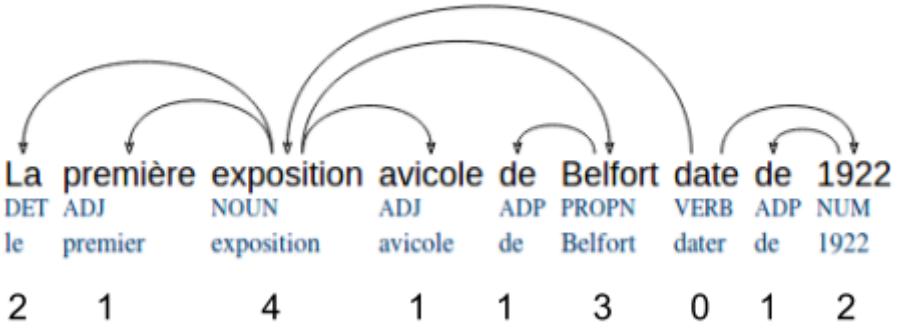


Figure 2-5

Si nous prenons un exemple venant de l'analyse de Universal Dependencies dans la figure 2-5, la valeur de MDD sera calculée comme : (2+1+4+1+1+3+0+1+2)/8= 1.875

Comparant la distance de dépendance dans des treebanks d'une version naturelle et deux versions avec un ordre des mots aléatoire (ces deux versions sont appelées *RL1* et *RL2* dans Liu (2008)) pour chaque langue, les résultats prouvent que la distance de dépendance moyenne est plus courte que celle de *RL1* et de *RL2* dans chacune des 20 langues; Le *RL2* dont les dépendances n'acceptent pas le croisement afin de respecter la projectivité, est plus court que le RL1. En conclusion, la complexité de la phrase n'est pas dûe à sa longueur, mais à sa distance de dépendance moyenne; La distance de dépendance a une tendance à se minimiser. Nous pouvons aussi retrouver une pensée similaire chez Tesnière (1965) : "*Le principe fondamental de la transformation de l'ordre structural en ordre linéaire est de transporter les connexions de l'ordre structural en séquences de l'ordre linéaire, de façon que les éléments qui sont en connexion dans l'ordre structural se trouvent en voisinage immédiat sur la chaîne parlée*" ; La grammaire, les contraintes formelles (projectivité) et la capacité cognitive travaillent ensemble pour garder la moyenne de distance de dépendance d'une langue dans le seuil.

Les travaux de Richard Futrell & al. (2015) avancent une notion similaire à 'la longueur de dépendance' avec l'hypothèse de DLM (*Dependency length minimization*) : *"language users prefer word orders that minimize dependency length. The hypothesis makes two broad predictions. First, when the grammar of a language provides multiple ways to express an idea, language users would prefer the expression with shortest dependency length. Indeed, speakers of a few languages have been found to prefer word orders with short dependencies*

when multiple options are available. Second, grammars should facilitate the production of short dependencies by not enforcing word orders with long dependencies" (Richard Futrell & al. ,2015)

Dans Richard Futrell & al. (2015), la longueur de dépendance est étudiée dans 37 langues, avec une attention portée sur l'ordre des mots. Leurs résultats prouvent que les locuteurs préfèrent universellement un ordre de mot qui pourrait minimiser la longueur de dépendance.

2.4.4 Flux de dépendance

La recherche de Liu (2008) et de Futrell et al. (2015) s'intéresse à la valeur du nombre de token entre le dépendant en question et son gouverneur. Le nombre de dépendances traversées une position entre deux mots pourrait probablement être une autre propriété contrôlée par la mémoire immédiate.

Ainsi, une étude quantitative basée sur des caractéristiques du flux de dépendance a été proposée :

Rappelons la définition et la taille du flux:

Le flux de dépendance en une position donnée (entre deux mots d'une phrase) est l'ensemble des dépendances qui relient un mot à gauche de cette position à un mot à droite. (Kahane 2001)

L'exemple du flux (lignes pointillées rouges):

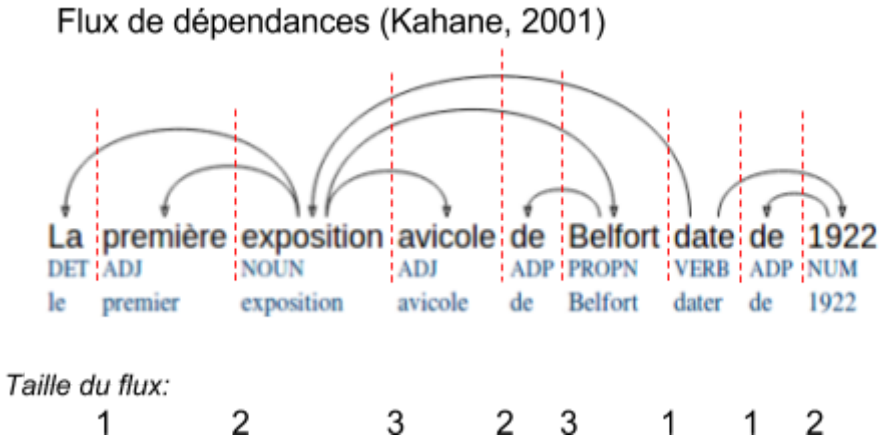


Figure 2-6

Dans la figure 2-6 les lignes en pointillée rouge représentent les positions inter-mot. Le flux est l'ensemble des dépendances qui coupe cette ligne, avec en-dessous la taille du flux

correspondante. La taille d'un flux est le nombre de dépendance du flux à la position en question. C'est l'information la plus basique concernant le flux. Par exemple, la position entre "la" et "première" n'est traversée que par une seule relation. La taille de flux à cette position est donc de 1. La taille du flux à la position entre "exposition" et "avicole" est de 3, car elle est traversée par 3 relations.

Les premiers travaux quantitatifs sur le flux de dépendance sont ceux de Jardonnet (2009), et Botalla, (2014):

- Jardonnet (2009) montre que la taille du flux dans un corpus de français écrit (le French Treebank en dépendance (Candito M.-H., 2009)), correspond bien à la limite cognitive de 7±2 pour la mémoire immédiate après la suppression des relations de ponctuation et l'aplatissement des dépendances en bouquet de certaines constructions comme coordination.
- Dans Botalla (2014), la projectivité et les diverses configurations de la langue française sont étudiées à travers le concept de flux de dépendance. Le travail d'investigation est établi sur le corpus de français parlé Rhapsodie (Lacheret et al., 2014; Kahane et al., 2013). Une représentation du flux en matrice est proposée. C'est la méthode que nous allons utiliser pour manipuler facilement les différents calculs des propriétés du flux.

Nous passons au chapitre suivant pour la représentation du flux et leurs propriétés.

Chapitre 3. Flux

3.1 Représentation en matrice

Nous suivons la représentation du flux en matrice (Botalla, 2014). Cette représentation nous permet non seulement une illustration lisible par l'humain, mais aussi une facilité de calcul par la machine.

"*La matrice se présente sous la forme d'un tableau, dans lequel les lignes représentent la suite des mots (qui sont l'extrémité d'un élément du flux) à gauche de la position et les*

colonnes la suite des mots (qui sont l'extrémité d'un élément du flux) à droite." Botalla (2014)

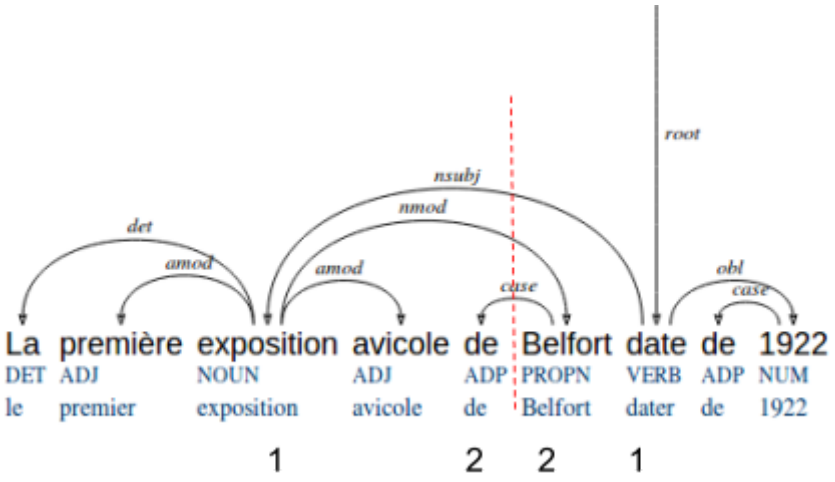


Figure 3-1

Nous retrouvons l'exemple précédent sous le format Universal Dependencies 2.0 (Nivre et al. 2016).

L'exemple du flux à la position entre 'de' et 'Belfort' représenté sous la forme d'une matrice :

	1	2
1	[VERB , NOUN , nsubj , -]	[NOUN , PROPN , nmod , +]
2		[PROPN , ADP , case , -]

Figure 3-2

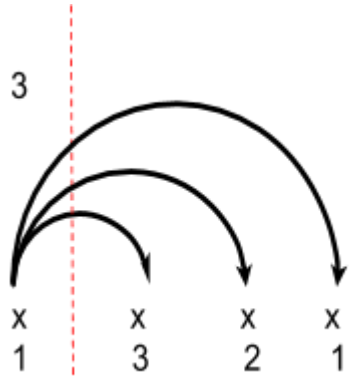
La première case : [VERB , NOUN , nsubj , -] représente la relation *nsubj* entre "date" et "exposition". Les deux premiers éléments sont l'étiquette de catégorie morpho-syntaxique de gouverneur (VERB pour le mot "date") et de dépendant (NOUN pour le mot "exposition"), le troisième élément est le nom de la relation *nsubj*, le quatrième élément est la direction de la dépendance : "-" pour le cas du dépendant gauche et "+" pour le cas du dépendant droit.

Dépendances en bouquet dans la matrice

Un bouquet est un ensemble de dépendances partageant le même sommet. Le bouquet est dit *right-branching* lorsque le sommet commun est sur la gauche, et *left-branching* lorsque le sommet commun est à droite.

Dans la matrice les relations dans une même ligne ou dans une même colonne sont des relations en bouquet. Comme la figure 3-3-1 , les relations en bouquet à gauche sont en une seule ligne dans la matrice, et la figure 3-3-2 montre les relations en bouquet à droite qui sont en une seule colonne dans la matrice.

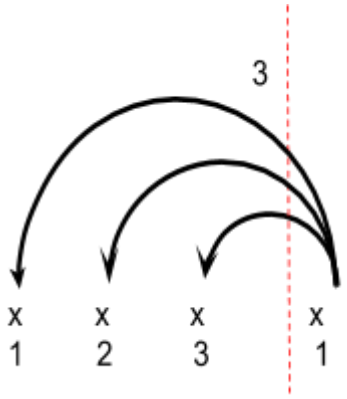
Bouquet à gauche :



	1	2	3
1	X	X	X

Figure 3-3-1

Bouquet à droite :



	1
1	X
2	X
3	X

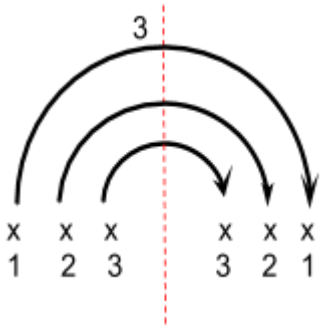
Figure 3-3-2

Reprenons l'exemple de la figure 3-2, il s'agit d'un bouquet à gauche : la relation { exposition < **nsubj** date} et {exposition **nmod** > Belfort} mais également d'un bouquet à droite pour la relation {exposition **nmod** > Belfort} et {de < **case** Belfort}.

3.2 Propriétés

3.2.1 Dépendances disjointes

Dépendances sont dites *disjointes* si elles ne partagent aucun sommet (Botalla, 2014). La figure 3-4 nous montre un exemple avec trois dépendances disjointes dans un même flux. Dans la figure 3-1 , il y a 2 dépendances disjointes : {de < **case** Belfort} et { exposition < **nsubj** date} dans le flux entre "de" et "Belfort".



	1	2	3
1	X		
2		X	
3			X

Figure 3-4

Regardons l'exemple artificiel contenant les constructions d'auto-enchâssement dans la section 2.4.1 :

Le chat que le chien que l'homme aime poursuit miaule.

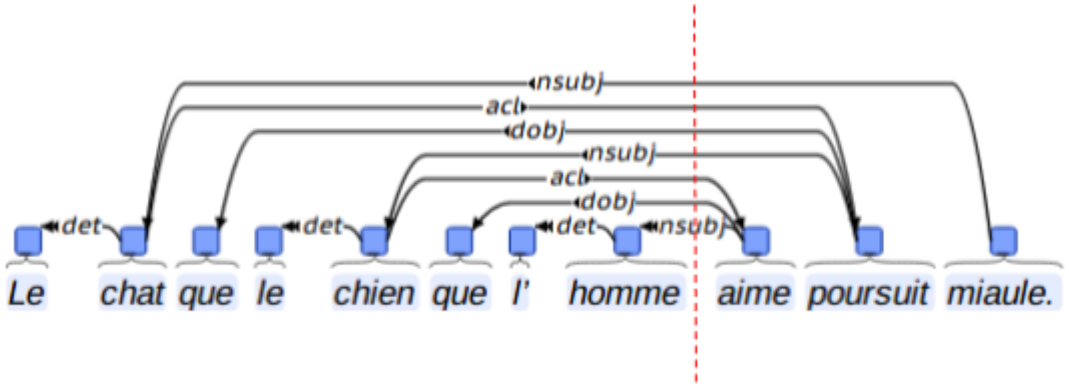


Figure 3-5

À la position entre "homme" et "aime", nous pouvons constater que les trois relations *nsubj* sont des dépendances disjointes, ceci correspond aussi au nombre d'auto-enchâssement dans l'analyse de la section 2.4.1: le niveau 0 est la relation sujet entre "chat" et "miaule", le niveau 1 est celle d'entre "chien" est "poursuit", le niveau 2 est celle d'entre "homme" et "aime". Ainsi, le nombre de dépendances disjointes peut mesurer les niveaux d'auto-enchâssement.

Le flux contenant les dépendances disjointes sont visuellement faciles à repérer sur la représentation en matrice : deux dépendances sont disjointes si elles ne sont ni sur la même colonne, ni sur la même ligne. En particulier les dépendances sur la diagonale sont disjointes les unes par rapport aux autres.

3.2.2 Poids du flux

Le *poids* du flux (angl. *flux weight*) est la taille du plus grand sous-flux disjoint. Si nous prenons l'exemple de la figure 3-4, la taille du plus grand sous-flux disjoint est de 3 (il s'agit

de 3 relations disjointes au maximum). Ainsi, le poids du flux en question est de 3. Si nous prenons l'exemple de la figure 3-1, il s'agit du plus grand sous-flux disjoint de valeur 2 : la relation *case* et la relation *nsubj*, donc le poids du flux en question est de 2. Pour la figure 3-5, la taille du flux est de 7; Pour les figure 3-1 et 3-4, cette valeur est de 3. La taille du flux ne permet pas de révéler si les dépendances partagent un même sommet, par conséquent elle ne peuvent pas rendre compte de phénomènes particuliers comme l'auto-enchâssement. Pour un flux contenant 3 relations en bouquet, c'est à dire 3 relations qui partagent un sommet d'un coté gauche ou droit, la taille du flux est évidemment 3, ce qui est identique avec un flux ayant 3 relations disjointes. Pourtant, le poids est seulement de 1 pour un flux contenant 3 relations en bouquets, et le poids est de 3 pour un flux contenant 3 relations disjointes.

La représentation du flux en matrice nous permet de repérer également le poids. Ce qui différencie du flux ayant exclusivement des dépendances disjointes, est que pour un flux mélangé de dépendances en bouquet et de dépendances disjointes le flux est plus compliqué à calculer.

	1	2	3	4
1	X1	X2	X3	
2			X4	
3			X5	X6

Figure 3-6

Nous pouvons compter la valeur maximale de son sous-flux disjoints comme ce que la définition du poids du flux suggère :

Prenons la matrice de la figure 3-6, nous trouvons deux sous-flux disjoints de taille 3 étant la valeur maximale:

d1 = X1, X4, X6

(X1,X4 et X6 ne sont ni sur la même colonne ni sur la même ligne)

d2 = X2, X4, X6

(X2, X4 et X6 ne sont ni sur la même colonne ni sur la même ligne)

Cela nous permet de bien avoir la valeur du poids qui est de 3. Concernant le calcul par la machine, nous allons présenter un algorithme dans le chapitre 5.

3.2.3 Densité

La densité est la proportion entre la taille du flux et le poids du flux, autrement dit, elle nous indique la présence de bouquet dans le flux.

Pour calculer la densité , nous avons :

Densité = Poids/Taille

Dans l'exemple de la figure 3-1, la densité est de 2/3. Pour un flux ayant exclusivement 3 dépendances disjointes, la densité est 3/3=1, ceci est la valeur maximale de la densité, indiquant qu'il n'y a aucun bouquet.

3.2.4 Empan

L'empan (angl. *span*) est également une caractéristique du flux. L'empan gauche du flux est le nombre de mots à gauche, et l'empan droit du flux est le nombre de mots à droite. L'empan gauche correspond au nombre de mots en attente d'un gouverneur ou d'un dépendant à droite de la position du flux, l'empan droit correspond au nombre d'éléments attendus. Cette propriété peut être théoriquement un indice à repérer si un bouquet de dépendances branche à droite ou à gauche: le sommet commun à gauche (empan droit > empan gauche); le sommet commun à droite (empan gauche > empan droit).

Dans la représentation du flux en matrice, l'empan gauche du flux correspond au nombre de lignes et l'empan droit correspond au nombre de colonnes. Dans la figure 3-3-1, l'empan gauche est 1, l'empan droit est 3. Dans la figure 3-3-2, l'empan gauche est 3, l'empan droit est 1. Cela correspond aux valeurs de largeur et de longueur dans leurs matrices .

3.2.5 Rapport D/G

Le rapport D/G est la proportion entre l'empan droit et l'empan gauche. Cette proportion nous indique si un flux de dépendance est de type *right-branching* ou bien de type *left-branching*. Pour calculer rapport D/G, nous avons :

Rapport D/G = empan droit / empan gauche = nombre de lignes / nombre de colonnes (dans la représentation matricielle)

Ainsi, pour un flux de dépendance en *right-branching*, la valeur du rapport D/G est de moins de 1, et pour un flux de dépendance en *left-branching*, la valeur de D/G est de plus de 1.

3.3 Hypothèses

Nous avons présenté dans la section précédente des propriétés du flux : la taille du flux, le poids du flux, la densité, l'empan et le rapport D/G. Nos hypothèses sont ainsi lancées comme ci-dessous:

Certains pourraient penser que le flux représente l'ensemble des relations syntaxiques en attente que le locuteur devrait garder dans sa mémoire après avoir produit chaque mot, et que la taille du flux devrait être limitée par la même frontière que celle indiquée par Miller (1956) et ne pas dépasser 7 ± 2. Pourtant, il nous semble que les locuteurs peuvent factoriser l'information des bouquets, et donc notre hypothèse est que ce qui compte c'est le nombre de dépendances disjointes (c'est-à-dire le poids) et non la taille du flux.

Quant au poids du flux, il nous permettrait de rendre compte de la difficulté cognitive sur une construction en bouquet et une construction auto-enchâssée (Étant donné qu'un bouquet de dépendances a du poids à 1, nous supposons qu'il prend seulement une place dans la mémoire immédiate peu importe sa taille de flux), notre hypothèse est que le poids est une bonne mesure de la complexité de la phrase, et nous allons voir s'il s'agit d'une limitation et sa relation avec la frontière de 7 ± 2 de Miller (1956).

La densité nous aidera à vérifier la tendance de choix entre construction d'auto-enchâssement (en dépendances disjointes) et construction en dépendances en bouquet. La densité pourrait avoir une valeur basse si nous préférons les constructions en bouquet aux constructions auto-enchâssées. Nous allons donc comparer ces valeurs dans tous les treebanks UD et vérifier si ces valeurs sont basses.

Concernant l'empan et le rapport D/G, l'hypothèse est qu'ils pourraient nous aider à repérer les différents types de langues. Du point de vue de la typologie des langues: les langues à tête initiale (la tête tend à se placer à droite de son dépendant) ont des arbres *right-branching*, ex: l'arabe standard ou le gallois; Les langues à tête finale (la tête tend à se placer à gauche de son dépendant) ont des arbres *left-branching*, ex: le japonais, le coréen, ou le turc. Comme ce qu'on a expliqué dans la section précédente à propos de la valeur de rapport D/G, nous voudrions vérifier si le rapport D/G est de plus de 1 pour les langues à tête initiale, et vice versa, mais nous verrons que ce n'est pas le cas avec l'annotation de UD.

Chapitre 4 Treebank UD version 2.0

4.1 Présentation du projet UD (Universal dependencies treebank)

Le schéma UD est actuellement le schéma d'annotation syntaxique de référence en traitement automatique des langues (NLP, natural language processing), principalement basé sur le schéma de Universal Stanford Dependencies (de Marneffe et al., 2014) et de Google universal part-of-speech tags (Petrov et al., 2012). La bibliothèque de UD nous permet de faire des études sur plus de 50 langues à partir de ses données annotées en arbre de dépendance.

4.2 Schéma UD version 2.0

4.2.1 Syntaxe

La taxonomie des relations de dépendance syntaxique a été renouvelée dans la deuxième version du schéma UD, il s'agit au total de 37 relations universelles pour toutes les langues, par rapport à la première version du schéma UD dont le nombre de relations était de 42 (Marneffe *et al.* 2014). A part des relations universelles, dans la deuxième version du schéma UD, différentes relations spécifiques sont possibles afin de bien annoter les structures spécifiques d'une langue.

Les relations universelles :

- acl: proposition adjectivale
- advcl: proposition adverbiale
- advmod: modifieur adverbial
- amod: modifieur adjectif
- appos: modifieur d'apposition
- aux: auxiliaire
- cas: marquage de cas
- cc: conjonction de coordination
- ccomp: complément propositionnel
- clf: classifieur

- compound: mot composé
- conj: conjunction
- cop: copule
- csubj: sujet propositionnel
- dep: dépendance non spécifiée
- det: déterminant
- discours: élément du discours
- dislocated: élément disloqué
- expl: expletive , élément nominal sémantiquement vide
- fixed: expression multi-mots figée
- flat: expression multi-mots plate
- gosewith: deux ou plusieurs parties d'un mot séparés dans un texte à cause d'une mauvaise tokenisation
- iobj: objet indirect
- list: liste
- mark: marqueur
- nmod: modifieur nominal
- nsubj: sujet nominal
- nummod: modifieur numérique
- obj: objet
- obl: oblique nominal
- orphan: orphelin
- parataxis: parataxis
- punct: ponctuation
- reparandum: disfluence surchargé
- root: racine
- vocatif: vocatif
- xcomp: complément verbal ou adjectival dont l'un des actants n'est pas réalisé localement

Les relations spécifiques sont sous forme *universal:extension* (*universal* représente la relation principale existant dans toutes les langues et *extension* est le sous-type que certaines langues développent dans leurs constructions spécifiques)

Par exemple, dans le schéma UD du chinois en version 2.0, il s'agit des différentes sous-types de *compound* (mot composé) pour le cas des verbes composés , les éléments dans ce genre de verbe sont syntaxiquement séparés et analysables bien qu'ils forment sémantiquement une seule unité sémantique :

- compound:dir : verbe composé en verbe + particule directionnel
- compound:ext : verbe composé en verbe + particule descriptive
- compound:vo : verbe composé en verbe + objet
- compound:vv : verbe + verbe (construction de verbes en série (angl. *serial verb construction*))

Certaines langues peuvent ne pas posséder certaines relations universelles, comme par exemple la relation *clf* (classifieur) qui n'est pas présente dans la langue française, mais fréquente dans la langue chinoise. Inversement, la relation *expl* (explétive) n'existe pas dans la langue chinoise mais elle existe dans la langue française : ex. la relation avec *il* dans *il pleut*.

4.2.2 Morphologie

Dans le schéma UD, on associe à chaque token un lemme, une étiquette morpho-syntaxique et des propriétés grammaticales. Pour ces deux derniers, il s'agit des parties du discours (angl. *part-of-speech* (*POS*)) et des traits *features*.

Les *POS* sont universelles :

- ADJ: adjectif
- ADP: adposition
- ADV: adverbe
- AUX: auxiliaire
- CCONJ: conjonction de coordination
- DET: déterminant
- INTJ: interjection
- NOM: nom
- NUM: numéral

- PARTIE: particule
- PRON: pronom
- PROPN: nom propre
- PUNCT: ponctuation
- SCONJ: conjonction subordonnée
- SYM: symbole
- VERB: verbe
- X: autre

Les traits (angl. *features*) sont universels et spécifiques:

- Abbr: abréviation
- Animacy: animéité
- Aspect: aspect
- Case: cas
- Definite: defini ou état
- Degree: degré de comparaison
- Evident: évidence
- Foreign: mot étranger
- Gender: genre
- Mood: mode
- NumType: type numéral
- Number: nombre
- Person: personne
- Polarity: polarité
- Polite: politesse
- Poss: possessif
- PronType: type pronominal
- Reflex: réfléchi
- Tense: temps
- VerbForm: forme du verbe ou du deverbatif
- Voice: voix

4.3 Annotation de construction bouquet *vs* chaine

Dans le schéma UD, la coordination est annotée en bouquet (voir la figure 4-1). Cette annotation est différente de celle du corpus Rhapsodie (Lacheret et al. 2014, Kahane et al. 2012) et du corpus Orféo (Wang, Kahane & Telliers, 2014) où la coordination est annotée en chaîne comme la figure 4-2 (Gerdes & Kahane, 2009). Cette différence influence évidemment la taille du flux, pour la position marquée par ligne en pointillée rouge dans la figure 4-1 , la taille est de 3 , pour le cas dans la figure 4-2 la taille est de 1. L'annotation en bouquet augmente également la valeur de l'empan droit sans faire de différence entre langues à tête initiale ou à tête finale.

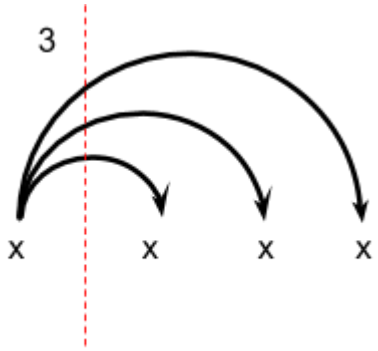


Figure 4-1

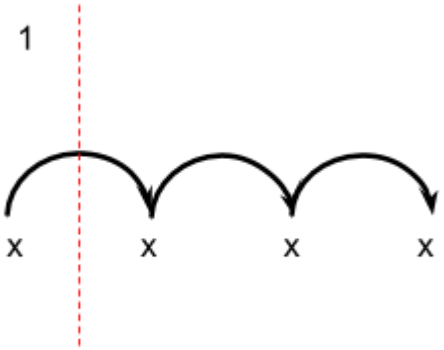


Figure 4-2

Nous proposons un exemple en anglais du corpus *UD_English-ParTUT* afin de bien montrer ces deux représentations dans la figure 4-3. La première phrase vient de UD original où la coordination est annotée en bouquet, la deuxième est le résultat de notre conversion en chaîne.

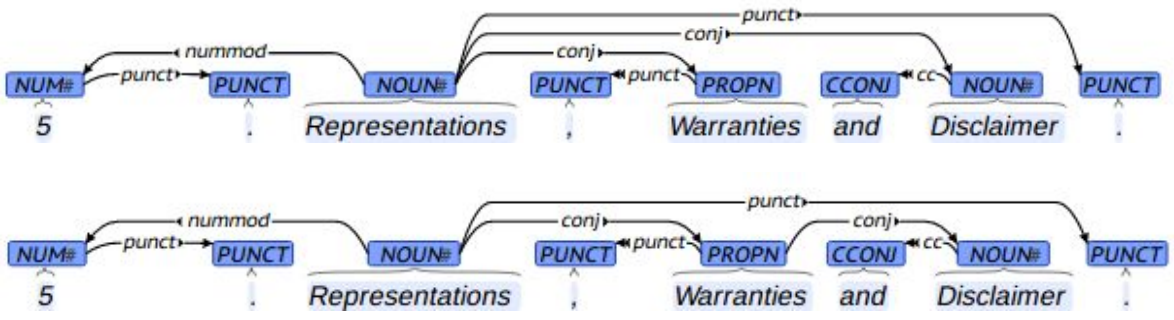


Figure 4-3 bouquet *vs* chaîne

Ainsi, à part les hypothèses faites dans chapitre 3 que nous voulons étudier, le deuxième travail que nous envisageons de faire est d'étudier les différences au niveau des propriétés du flux touchées par ces deux représentations.

Chapitre 5 Méthode

5.1 Expérience

Nous utilisons les 70 treebanks en 50 langues distribués par le projet UD version 2.0. Tous ces corpus ont été corrigés manuellement et basés sur un schéma d'annotation unifié bien que le développement des treebanks ait été fait par différents groupes.

Les 70 treebanks contenant au total 12 M mots et 630 K phrases :

UD_Ancient_Greek, UD_Ancient_Greek-PROIEL, UD_Arabic, UD_Arabic-NYUAD, UD_Basque, UD_Belarusian, UD_Bulgarian, UD_Catalan, UD_Chinese, UD_Coptic, UD_Croatian, UD_Czech, UD_Czech-CAC, UD_Czech-CLTT, UD_Danish, UD_Dutch, UD_Dutch-LassySmall, UD_English, UD_English-LinES, UD_English-ParTUT, UD_Estonian, UD_Finnish, UD_Finnish-FTB, UD_French, UD_French-ParTUT, UD_French-Sequoia, UD_Galician, UD_Galician-TreeGal, UD_German, UD_Gothic, UD_Greek, UD_Hebrew, UD_Hindi, UD_Hungarian, UD_Indonesian, UD_Irish, UD_Italian, UD_Italian-ParTUT, UD_Japanese, UD_Kazakh, UD_Korean, UD_Latin, UD_Latin-ITTB, UD_Latin-PROIEL, UD_Latvian, UD_Lithuanian, UD_Norwegian-Bokmaal, UD_Norwegian-Nynorsk, UD_Old_Church_Slavonic, UD_Persian, UD_Polish, UD_Portuguese, UD_Portuguese-BR, UD_Romanian, UD_Russian, UD_Russian-SynTagRus, UD_Sanskrit, UD_Slovak, UD_Slovenian, UD_Slovenian-SST, UD_Spanish, UD_Spanish-AnCora, UD_Swedish, UD_Swedish-LinES, UD_Tamil, UD_Turkish, UD_Ukrainian, UD_Urdu, UD_Uyghur, et UD_Vietnamese.

Notre expérience prévoit deux prétraitements à partir de ces 70 treebanks:

Premièrement, étant donné que la présence de la relation de ponctuation est une spécificité de l'écrit, et que la qualité de son annotation n'est pas satisfaisante, nous l'enlevons afin de rendre les différents corpus plus comparables. Cette version de corpus, nous l'appelons UD-original.

Ensuite, nous avons converti certaines relations annotées en bouquet en chaîne à partir de UD-original, cette version nous l'appelons UD-converti. Les relations que nous avons transformées sont :

conj, conj:discourse, conj:extend, conj:preconj, flat:foreign, flat:name, flat, fixed, appos, et orphan.

Dans UD-original et UD-converti, nous calculons les diverses propriétés du flux dans chacun des 70 treebanks : la taille du flux, le poids du flux, la densité, l'empan gauche et l'empan droit, et le rapport D/G.

5.2 Technique et algorithme

5.2.1 Technique

Le traitement informatique pour notre expérience se base sur Arborator (Gerdes, 2013) avec ses scripts en Python afin de traiter le treebank et d'établir les bases de données, et le projet de Bollata (2014) re-adapté pour l'objectif de calculer systématiquement dans notre projet les propriétés du flux à partir de la représentation du flux en matrice.

Au niveau technique, avant de commencer les calculs de notre expérience, il s'agit de trois étapes :

1. Prétraitement du corpus
2. Ecriture de base de données à l'aide d'Arborator
3. Traitement de la base de donnée afin de récupérer les dictionnaires contenant du flux représenté en matrice

5.2.2 Algorithme pour calculer le poids du flux

Une fois obtenu le dictionnaire de matrices de flux pour les 70 treebanks, nous pouvons commencer les calculs sur les propriétés du flux. Parmi ces propriétés, nous remarquons que le poids du flux est le plus compliqué à calculer. Nous proposons l'algorithme suivant :

Nous rappelons que le poids est la taille du plus grand sous-flux disjoint. Pour le calculer, nous pouvons commencer par une dépendance D dans le flux avec au moins un sommet qui n'est pas partagé avec d'autres dépendances dans le flux (une telle dépendance existe car la structure est acyclique). Ensuite, nous supprimons toutes les dépendances qui partagent un sommet avec D et ne peuvent donc pas être disjointes de D. Si le flux qui reste n'est pas vide, nous commençons par le même processus exactement : choisir une dépendance avec au moins un sommet qui n'est pas partagé avec d'autres dépendances dans le flux restant et en

supprimant toutes les dépendances partageant un sommet avec lui. À la fin, nous obtenons l'un des plus grands ensembles de dépendances disjointes dans le flux. Cet algorithme simple est en temps linéaire.

Pour avoir la valeur du poids d'un flux à partir d'une matrice, cet algorithme pourra être réalisé concrètement comme ci-dessous:

```
poids = 0
def fonction (matrice, poids):
    si la matrice n'est pas vide :
        pour chaque ligne non-vide (de la première jusqu'à la dernière ):
            pour chaque élément non-vide dans la ligne (du premier jusqu'au dernier):
                si un  élément qui est seul dans sa colonne:
                    choix=élément
                    poids+1
                    arrêt de cette condition
        si il n'existe pas de 'choix':
            pour chaque colonne non-vide (de la première jusqu'à la dernière ):
                pour chaque élément non-vide dans la ligne (du premier jusqu'au
            dernier):
                    si un élément est seul dans sa ligne:
                        choix=élément
                        poids+1
                        arrêt de cette condition
        supprimer la ligne et la colonnes où le choix se trouve
        relancer fonction avec la matrice et poids modifiée
    si la matrice est vide :
        return poids
```

Nous retrouvons l'exemple de matrice dans le chapitre 3 avec lequel nous montrons notre algorithme par étapes:

Étape A :

	1	2	3	4
1	**X1**	X2	X3	
2			X4	
3			X5	X6

1. la matrice n’est pas vide
2. X1 est le premier élément seul dans sa colonne
3. choisir X1
4. poids=poids+1=1
5. supprimer ligne 1 et colonne 1 où X1 se trouve

Étape B (les cases en gris sont supprimées à la fin de l’ étape A) :

	1	2	3	4
1	X1	X2	X3	
2			X4	
3			X5	**X6**

1. la matrice n’est pas vide
2. X6 est le premier élément qui est seul dans sa colonne
3. choisir X6
4. poids = poids +1 = 2
5. supprimer ligne 2 et colonne 3 où X6 se trouve

Étape C (les cases en gris sont supprimées à la fin de l’ étape A et B) :

	1	2	3	4
1	X1	X2	X3	
2			**X4**	
3			X5	X6

1. la matrice n’est pas vide

2. X4 est le premier élément qui est seul dans sa colonne
3. choisir X4
4. poids = poids +1 =3
5. supprimer ligne 1 et colonne 2 où X4 se trouve

Étape D :

1. la matrice est vide
2. sortir la valeur poids = 3

Chapitre 6 Résultats et discussion

6.1 Résultats généraux pour UD

Le tableau 6-1 nous montre les résultats de nos calculs sur les propriétés du flux, à partir du corpus UD-original.

Liste de mesures:

T-max : taille maximale du flux

T-M : taille moyenne du flux

P-max : poids maximal

P-M : moyenne du poids

E-G-max : empan gauche maximal

E-D-max : empan droit maximal

E-G-M : moyenne d'empan gauche

E-D-M : moyenne d'empan droit

D/G-M : moyenne de proportion d'empan droit/empan gauche

D-M : densité en moyenne

D'après les résultats dans le tableau 6-1, il n'y a pas de limite universelle de la taille du flux pour ces 50 langues. Nous avons constaté que les tailles maximales pour les 70 corpus varient à partir de 8 (*Kazakh, Sanskrit, Uyghur,* et *Vietnamese*) jusqu'à 97 (*Ancient-Greek*) , les tailles moyennes se distribuent entre 1.92 (*Polish*) et 3.61 (*Czech-CLTT*). La taille peut être

influencée par l'annotation des constructions en bouquet, notamment pour les relations de coordination (conj), d'expression multi-mots (*fixed*), d'expression multi-mots plate (*flat*) et d'apposition (*appos*).

Par contre, les poids du flux sont nettement plus stables. Les poids maximaux pour les 70 corpus sont entre 3 (*Sanskrit*) et 7 (*Czech-CLTT*) , et les moyennes du poids sont entre 1.18 (*Polish* et *Slovak*) et 1.77 (*Czech-CLTT*). Afin d'examiner notre hypothèse concernant le poids du flux, nous allons l'étudier plus précisément dans le tableau 6-2.

En ce qui concerne l'empan, l'empan gauche est plus stable que l'empan droit. Nous nous intéressons particulièrement au rapport entre l'empan droit et l'empan gauche (rapport D/G), En observant les rapports D/G les plus hauts, nous attendons théoriquement les langues à tête initiale : nous avons notamment 1.32 pour *Arabic*, 1.55 pour *Arabic-NYUAD*, 1.36 pour *Czech-CLTT*, 1.31 pour *Old church Slavonic* et 1.37 pour *Irish*. La langue arabe, la langue vieux slave d'église et la langue irlandaise sont typiquement des langues à tête initiale et nos résultats le prouvent bien. Quant à la langue tchèque , il s'agit de deux autres corpus ayant la proportion de 1.00 et de 1.03, donc nous considérons que le rapport de 1.36 pour *Czech-CLTT* contenant 814 phrases n'est pas forcément significatif. Les résultats des langues à tête finale ne sont pas pertinents, par exemple pour le cas de japonais qui a 1.17, tandis que le rapport minimal est de 0.77, et la moyenne de rapport pour tout corpus UD-original est de 1.05.

La densité varie entre 57.00 % (*Persian*) et 72.20 % (*Polish*), pour tout corpus UD-original la densité est de 65.31%. En fait, beaucoup de flux ayant la densité 1 est un flux de taille 1 ayant une seule dépendance.

6.2 Poids

Le tableau 6-2 nous montre les résultats en pourcentage par position pour le poids de 1 à 6 dans les 70 corpus pour les 50 langues: dans tous les corpus le poids atteint rarement 6 : 0.05% des positions pour *Czech-CLTT*, 0.02% pour *Hungarian* et 0.01% pour *Chinese* et *Latvian,* 0.00% pour tous les autres .

Nous avons remarqué également que la taille du corpus pourrait rendre le corpus exceptionnel, comme ce qui indique dans les trois corpus du thèque, le corpus *Czech-CLTT* contient 814 phrases, cela rend les calculs moins stables que les deux autres corpus ayant des

pourcentages assez proches. Si on met de côté les corpus ayant moins de 1000 phrases (notamment *Czech-CLTT* et *Uyghur*), nous constatons que les langues où le poids est le plus important sont arabe, chinois et coréen: plus de 10 % pour le poids de 3. Nous avons observé que les corpus où le poids est le moins important sont *Finnish-FTB*, *Polish*, et *Slovak* : plus de 80 % pour le poids de 1.

En ce qui concerne tout corpus UD-original (montré dans la ligne 'total' du tableau), les flux ayant du poids à taille 1 sont majoritaires (62.15 %), 99.62% des positions flux ont un poids inférieur ou égale à 3, il s'agit de seulement 0.36% pour un poids de 4, 0.02% pour un poids de 5 et 0.00% pour un poids de 6.

Quant à la comparaison au sein d'une même langue dans les corpus différents : grec ancien, arabe, tchèque, néerlandais, anglais, finnois, français, galicien, italien, norvégien, latin, slovaque, portugais, russe, espagnol, et suédois, notamment pour les corpus du grec ancien, du norvégien, de l'arabe, du portugais et du russe, nous constatons une forte tendance similaire dans les résultats. Concernant les autres langues, les divergences dans la même langue seraient probablement dues au genre du corpus ou au choix d'annotation assez différents.

	Tokens	Arbres	T-max	T-M	P-max	W-av	E-G-max	E-D-max	E-G-M	E-D-M	D/G-M	D-M
UD_Ancient_Greek	182030	12613	97	3.01	6	1.49	12	97	2.31	1.99	1.13	60.32%
UD_Ancient_Greek-PROIEL	198034	15865	31	2.89	6	1.49	12	29	2.19	1.99	1.14	61.96%
UD_Arabic	254120	6984	36	2.93	5	1.66	9	35	2.06	2.41	1.32	66.47%
UD_Arabic-NYUAD	738889	19738	78	3.12	6	1.66	12	78	1.95	2.74	1.55	64.65%
UD_Basque	97069	7194	13	2.25	5	1.36	9	11	1.86	1.63	1.05	70.68%
UD_Belarusian	6864	333	17	2.48	4	1.44	9	17	1.98	1.78	1.09	69.28%
UD_Bulgarian	140425	10022	14	2.24	5	1.28	9	14	1.90	1.50	0.97	67.67%
UD_Catalan	474069	14832	20	2.69	6	1.48	13	19	2.17	1.83	1.03	64.16%
UD_Chinese	111271	4497	27	3.24	6	1.65	14	25	2.77	1.86	0.84	61.28%
UD_Coptic	8519	320	9	2.74	4	1.43	8	8	2.23	1.74	1.00	60.08%
UD_Croatian	183816	8289	13	2.52	5	1.40	11	13	2.13	1.65	0.98	65.74%
UD_Czech	1332566	77765	56	2.43	6	1.37	17	56	2.03	1.63	1.00	67.23%
UD_Czech-CAC	483520	24081	47	2.50	6	1.39	11	47	2.04	1.71	1.03	66.49%
UD_Czech-CLTT	26781	814	28	3.61	7	1.77	10	24	2.36	2.83	1.36	62.24%
UD_Danish	90710	4947	16	2.61	4	1.34	16	12	2.20	1.61	0.97	63.32%
UD_Dutch	197925	13050	15	2.89	5	1.43	12	15	2.46	1.69	0.92	60.44%
UD_Dutch-LassySmall	91793	6841	29	2.74	4	1.33	10	29	2.06	1.87	1.21	61.32%
UD_English	229733	14545	18	2.58	6	1.35	13	17	2.19	1.58	0.92	63.01%
UD_English-LinES	67197	3650	25	2.54	5	1.35	10	24	2.13	1.61	0.95	63.89%
UD_English-ParTUT	38114	1590	15	2.63	5	1.39	9	14	2.26	1.60	0.89	62.79%
UD_Estonian	34628	3172	10	2.26	5	1.25	9	10	1.82	1.58	1.11	68.04%
UD_Finnish	180911	13581	33	2.31	6	1.31	10	33	1.89	1.63	1.06	68.53%
UD_Finnish-FTB	143326	16856	14	2.06	5	1.19	11	14	1.77	1.39	0.98	70.29%
UD_French	392230	16031	34	2.51	5	1.39	11	34	2.04	1.71	1.02	65.09%
UD_French-ParTUT	17927	620	11	2.70	5	1.44	11	10	2.31	1.68	0.91	62.86%
UD_French-Sequoia	60574	2643	31	2.63	5	1.44	12	31	2.15	1.75	1.00	64.58%
UD_Galician	109106	3139	15	2.56	5	1.41	11	15	2.04	1.80	1.08	64.54%
UD_Galician-TreeGal	15436	600	13	2.55	4	1.43	9	12	2.12	1.71	1.00	65.30%
UD_German	281974	14917	28	3.00	6	1.46	13	26	2.51	1.76	0.96	59.84%
UD_Gothic	45138	4372	21	2.53	4	1.38	10	20	1.87	1.91	1.23	65.96%
UD_Greek	51351	2065	13	2.51	5	1.41	10	9	2.12	1.65	0.95	65.57%
UD_Hebrew	149088	5725	62	2.56	5	1.48	11	61	2.01	1.86	1.11	66.99%
UD_Hindi	316274	14963	18	3.20	6	1.58	13	15	2.76	1.84	0.85	59.67%
UD_Hungarian	31584	1351	13	2.83	6	1.54	10	10	2.44	1.75	0.89	64.54%
UD_Indonesian	110143	5036	28	2.31	5	1.39	9	28	1.75	1.85	1.22	70.30%
UD_Irish	13826	566	18	2.88	5	1.56	7	18	1.94	2.34	1.37	64.95%
UD_Italian	282611	13402	35	2.50	5	1.39	10	34	2.10	1.65	0.96	65.69%
UD_Italian-ParTUT	42651	1590	14	2.59	5	1.43	9	14	2.20	1.66	0.93	64.46%
UD_Japanese	173458	7675	15	2.79	5	1.55	15	11	2.17	2.03	1.17	64.52%
UD_Kazakh	529	31	8	2.67	4	1.52	6	5	2.21	1.82	1.00	67.07%
UD_Korean	63426	5350	23	2.73	5	1.62	9	20	2.25	1.93	0.99	68.80%
UD_Latin	18184	1334	17	2.86	5	1.52	8	16	2.31	1.87	1.02	63.32%
UD_Latin-ITTB	280734	16508	11	2.67	6	1.46	10	10	2.30	1.65	0.89	64.10%
UD_Latin-PROIEL	159407	15324	28	2.77	6	1.47	14	28	2.15	1.91	1.12	64.14%
UD_Latvian	44795	3054	18	2.48	6	1.39	9	17	2.04	1.68	0.99	67.31%
UD_Lithuanian	5356	263	14	2.43	4	1.38	9	13	2.06	1.60	0.95	68.01%
UD_Norwegian-Bokmaal	280256	18106	38	2.44	5	1.30	11	38	2.08	1.54	0.96	64.18%
UD_Norwegian-Nynorsk	276580	16064	38	2.50	6	1.32	11	38	2.12	1.57	0.96	63.82%
UD_Old_Church_Slavonic	47532	5196	20	2.48	5	1.34	8	19	1.76	1.93	1.31	66.03%
UD_Persian	136896	5397	14	3.45	6	1.64	13	10	3.03	1.81	0.77	57.00%
UD_Polish	72763	7127	10	1.92	4	1.18	8	7	1.62	1.40	1.04	72.20%
UD_Portuguese	217591	8891	19	2.54	5	1.43	13	19	2.12	1.70	0.98	65.84%
UD_Portuguese-BR	287884	10874	38	2.54	5	1.45	10	38	2.05	1.77	1.04	66.46%
UD_Romanian	202187	8795	14	2.39	6	1.40	9	14	1.95	1.69	1.05	67.75%
UD_Russian	87841	4429	31	2.34	5	1.37	10	30	1.83	1.74	1.12	69.62%
UD_Russian-SynTagRus	988460	55398	18	2.34	6	1.37	10	17	1.94	1.63	1.01	68.64%
UD_Sanskrit	1206	190	8	2.23	3	1.29	6	5	2.05	1.39	0.82	68.76%
UD_Slovak	93015	9543	10	2.00	4	1.18	9	8	1.74	1.36	0.96	70.22%
UD_Slovenian	126593	7212	17	2.50	5	1.30	13	17	2.21	1.47	0.87	64.12%
UD_Slovenian-SST	19488	2137	14	2.77	4	1.33	12	8	2.34	1.62	0.94	60.51%
UD_Spanish	419587	15587	38	2.51	5	1.42	11	38	2.03	1.74	1.04	65.91%
UD_Spanish-AnCora	496953	15959	31	2.63	5	1.47	12	31	2.16	1.76	1.00	65.21%
UD_Swedish	76442	4807	31	2.58	5	1.32	10	31	2.07	1.68	1.04	62.95%
UD_Swedish-LinES	64787	3650	25	2.55	5	1.34	10	24	2.07	1.67	1.03	63.25%
UD_Tamil	9581	600	10	2.42	4	1.48	9	8	2.07	1.74	1.00	70.91%
UD_Turkish	48093	4660	13	2.44	6	1.48	9	13	2.00	1.77	1.04	71.20%
UD_Ukrainian	12846	863	11	2.19	4	1.27	8	9	1.85	1.49	0.98	69.22%
UD_Urdu	123271	4595	32	3.44	5	1.66	15	29	2.92	1.96	0.85	58.34%
UD_Uyghur	1662	100	8	2.93	5	1.73	7	6	2.75	1.80	0.77	67.31%
UD_Vietnamese	31799	2200	8	2.09	4	1.25	7	8	1.68	1.57	1.12	70.45%
Total	1.2E+07	630518	97	2.62	7	1.43	17	97	2.11	1.79	1.05	65.31%

Tableau 6-1 Résultats globaux pour UD-original

	Tokens	Arbres	1	2	3	4	5	6
UD_Ancient_Greek	182030	12613	57.77%	35.79%	5.96%	0.45%	0.02%	0.00%
UD_Ancient_Greek-PROIEL	198034	15865	57.81%	35.97%	5.74%	0.46%	0.02%	0.00%
UD_Arabic	254120	6984	47.15%	41.10%	10.60%	1.10%	0.05%	0.00%
UD_Arabic-NYUAD	738889	19738	47.16%	40.86%	10.67%	1.23%	0.08%	0.00%
UD_Basque	97069	7194	67.85%	28.28%	3.66%	0.21%	0.01%	0.00%
UD_Belarusian	6864	333	62.43%	31.70%	5.37%	0.50%	0.00%	0.00%
UD_Bulgarian	140425	10022	73.86%	24.20%	1.87%	0.06%	0.00%	0.00%
UD_Catalan	474069	14832	57.82%	36.58%	5.30%	0.30%	0.01%	0.00%
UD_Chinese	111271	4497	49.73%	37.35%	10.91%	1.78%	0.22%	0.01%
UD_Coptic	8519	320	61.76%	33.74%	4.37%	0.13%	0.00%	0.00%
UD_Croatian	183816	8289	64.46%	31.70%	3.66%	0.17%	0.01%	0.00%
UD_Czech	1332566	77765	66.78%	29.61%	3.42%	0.17%	0.01%	0.00%
UD_Czech-CAC	483520	24081	65.16%	30.82%	3.80%	0.22%	0.01%	0.00%
UD_Czech-CLTT	26781	814	42.78%	41.74%	12.19%	2.71%	0.53%	0.05%
UD_Danish	90710	4947	69.13%	27.85%	2.90%	0.12%	0.00%	0.00%
UD_Dutch	197925	13050	63.41%	31.12%	5.00%	0.44%	0.02%	0.00%
UD_Dutch-LassySmall	91793	6841	70.30%	27.04%	2.50%	0.15%	0.00%	0.00%
UD_English	229733	14545	68.45%	28.08%	3.29%	0.17%	0.00%	0.00%
UD_English-LinES	67197	3650	68.40%	28.16%	3.20%	0.23%	0.01%	0.00%
UD_English-ParTUT	38114	1590	64.99%	31.09%	3.77%	0.15%	0.00%	0.00%
UD_Estonian	34628	3172	77.26%	20.37%	2.21%	0.15%	0.01%	0.00%
UD_Finnish	180911	13581	72.60%	23.79%	3.28%	0.30%	0.03%	0.00%
UD_Finnish-FTB	143326	16856	82.77%	15.91%	1.22%	0.10%	0.00%	0.00%
UD_French	392230	16031	64.31%	32.23%	3.27%	0.17%	0.01%	0.00%
UD_French-ParTUT	17927	620	60.66%	34.99%	4.05%	0.26%	0.03%	0.00%
UD_French-Sequoia	60574	2643	61.25%	33.76%	4.69%	0.29%	0.01%	0.00%
UD_Galician	109106	3139	62.78%	33.73%	3.34%	0.14%	0.00%	0.00%
UD_Galician-TreeGal	15436	600	61.98%	33.75%	4.02%	0.25%	0.00%	0.00%
UD_German	281974	14917	59.60%	35.60%	4.46%	0.33%	0.01%	0.00%
UD_Gothic	45138	4372	65.83%	30.51%	3.46%	0.21%	0.00%	0.00%
UD_Greek	51351	2065	62.49%	33.73%	3.59%	0.19%	0.00%	0.00%
UD_Hebrew	149088	5725	58.04%	36.56%	5.17%	0.23%	0.00%	0.00%
UD_Hindi	316274	14963	49.02%	44.76%	5.65%	0.54%	0.03%	0.00%
UD_Hungarian	31584	1351	56.04%	35.24%	7.58%	0.99%	0.14%	0.02%
UD_Indonesian	110143	5036	64.82%	31.38%	3.61%	0.18%	0.01%	0.00%
UD_Irish	13826	566	53.11%	38.57%	7.47%	0.82%	0.03%	0.00%
UD_Italian	282611	13402	64.93%	31.55%	3.37%	0.16%	0.01%	0.00%
UD_Italian-ParTUT	42651	1590	61.56%	34.48%	3.78%	0.17%	0.00%	0.00%
UD_Japanese	173458	7675	50.98%	42.82%	6.03%	0.17%	0.00%	0.00%
UD_Kazakh	529	31	55.27%	37.42%	6.88%	0.43%	0.00%	0.00%
UD_Korean	63426	5350	51.30%	37.10%	10.24%	1.28%	0.09%	0.00%
UD_Latin	18184	1334	56.34%	35.90%	7.01%	0.69%	0.06%	0.00%
UD_Latin-ITTB	280734	16508	60.44%	33.25%	5.85%	0.45%	0.02%	0.00%
UD_Latin-PROIEL	159407	15324	61.30%	31.41%	6.27%	0.92%	0.10%	0.00%
UD_Latvian	44795	3054	67.21%	27.51%	4.75%	0.48%	0.05%	0.01%
UD_Lithuanian	5356	263	66.76%	28.97%	4.07%	0.21%	0.00%	0.00%
UD_Norwegian-Bokmaal	280256	18106	72.73%	24.95%	2.23%	0.08%	0.00%	0.00%
UD_Norwegian-Nynorsk	276580	16064	70.73%	26.67%	2.50%	0.10%	0.00%	0.00%
UD_Old_Church_Slavonic	47532	5196	69.31%	27.41%	3.15%	0.13%	0.00%	0.00%
UD_Persian	136896	5397	45.75%	45.14%	8.52%	0.59%	0.01%	0.00%
UD_Polish	72763	7127	82.46%	16.96%	0.57%	0.00%	0.00%	0.00%
UD_Portuguese	217591	8891	61.69%	33.94%	4.16%	0.21%	0.01%	0.00%
UD_Portuguese-BR	287884	10874	60.06%	35.50%	4.24%	0.20%	0.01%	0.00%
UD_Romanian	202187	8795	64.42%	31.62%	3.74%	0.22%	0.00%	0.00%
UD_Russian	87841	4429	66.76%	29.75%	3.28%	0.21%	0.00%	0.00%
UD_Russian-SynTagRus	988460	55398	67.30%	29.04%	3.42%	0.23%	0.01%	0.00%
UD_Sanskrit	1206	190	71.95%	27.07%	0.98%	0.00%	0.00%	0.00%
UD_Slovak	93015	9543	83.09%	16.24%	0.66%	0.01%	0.00%	0.00%
UD_Slovenian	126593	7212	72.12%	25.49%	2.31%	0.08%	0.00%	0.00%
UD_Slovenian-SST	19488	2137	70.09%	26.59%	3.17%	0.15%	0.00%	0.00%
UD_Spanish	419587	15587	61.54%	34.75%	3.56%	0.15%	0.01%	0.00%
UD_Spanish-AnCora	496953	15959	58.20%	36.45%	5.10%	0.25%	0.00%	0.00%
UD_Swedish	76442	4807	70.77%	26.62%	2.50%	0.10%	0.01%	0.00%
UD_Swedish-LinES	64787	3650	69.32%	27.51%	3.03%	0.13%	0.00%	0.00%
UD_Tamil	9581	600	58.14%	36.18%	5.23%	0.45%	0.00%	0.00%
UD_Turkish	48093	4660	60.71%	31.69%	6.69%	0.85%	0.06%	0.00%
UD_Ukrainian	12846	863	74.84%	23.60%	1.49%	0.06%	0.00%	0.00%
UD_Urdu	123271	4595	44.94%	45.09%	8.82%	1.04%	0.11%	0.00%
UD_Uyghur	1662	100	43.87%	42.16%	11.50%	2.12%	0.34%	0.00%
UD_Vietnamese	31799	2200	76.10%	22.71%	1.18%	0.01%	0.00%	0.00%
Total	1.2E+07	630518	62.15%	32.75%	4.71%	0.36%	0.02%	0.00%

Tableau 6-2 Poids du flux pour UD-original

6.3 Comparaison d'annotation bouquet *vs* chaîne

Dans cette partie, nous passons les même calculs pour UD-converti. UD-converti est une version convertie de UD-original, dans UD-converti nous avons transformé automatiquement les relations de coordination (et aussi les relations similaires) en bouquet en chaîne. Ces relations sont :

conj, conj:discourse, conj:extend, conj:preconj, flat:foreign, flat:name, flat, fixed, appos, et *orphan.*

La taille maximale du flux diminue jusqu'à 6 (*Sanskrit*) et 77 (*Arabic-NYUAD*), ainsi qu'entre 1.89 (*Polish*) et 3.44 (*Persan*) pour les moyennes de la taille du flux. Nous avons remarqué que pour le corpus Arabic-NYUAD, la taille maximal n'a diminué que de 1, après avoir vérifié dans le corpus, la phrase : # *sent_id = ANN20020815.0034:3* dans le fichier *ar_nyuad-ud-train.conllu,* contenant 385 tokens, les relations *nmod* forment un bouquet extrêmement grand, tous ces relations dépendent du noeud du numéro 5. Puisque les relations *nmod* ne faisaient pas partie de la liste de transformation en chaîne, ces relations restent toujours en bouquet. Pour le corpus de l'hébreu, la taille maximale est de 62 pour UD-original et aussi pour UD-converti, la phrase problématique est : *sent_id = 4897* dans le fichier *he-ud-train.conllu*, dans laquelle les relations *dep* forment un bouquet extrêmement grand, il paraît évident que cette annotation est une erreur (voir la figure 6-1), la relation *dep* ne doit être utilisée que lorsqu'aucune autre relation ne convient.

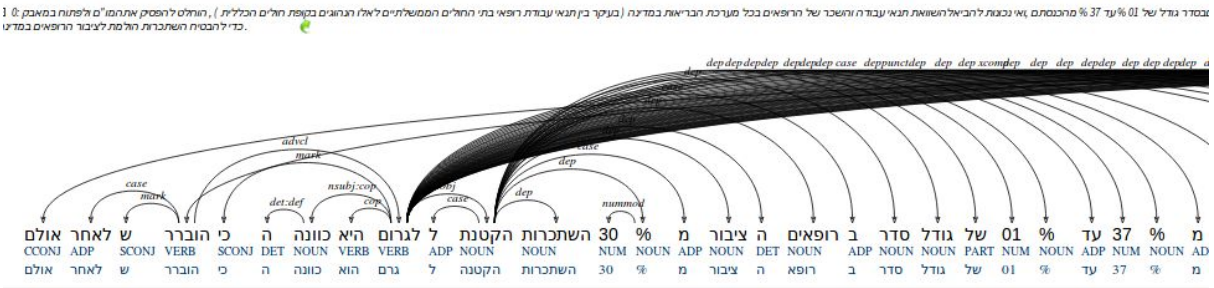


Figure 6-1

Selon le tableau 6-3, en observant les résultats de la taille maximale dans le corpus UD-converti, nous avons constaté que les tailles maximales ont principalement diminué.

En considérant les moyennes de la taille du flux, nous avons la même conclusion: les représentations de la chaîne nous permettent d'avoir un flux plus étroit.

Finalement, bien que les flux soient plus étroits dans UD-converti, notre transformation ne permet toujours pas de trouver une limite universelle pour la taille du flux dans les 50 langues, d'ailleurs, pour les autres propriétés comme l'empan et le poids, la différence entre les résultats de UD-original et de UD-converti n'est pas significative:

- Le poids moyen pour UD-converti complet est de 1.43, il est identique à celui de UD-original. Nous trouvons que la valeur du poids est quasiment indépendant de ces deux représentations.
- La moyenne de l'empan gauche pour UD-converti complet est de 2.11, valeur identique à celle de UD-converti. La moyenne de l'empan droit pour UD-converti complet est de 1.71, valeur inférieure à UD-original qui a 1.79. Notre conversion nous permet de diminuer l'empan droit, mais certaines relations restent difficiles de traiter comme celles que l'on a repérées ci-dessus, la relation *dep* ou bien la relation *nmod*.

Quant à la densité, la moyenne de UD-converti a augmenté d'environs 1% (66.47% contre 65.31%). Notre traitement, qui a transformé un certain nombre de représentations en bouquet, rend la densité plus grande.

	Tokens	Trees	T-max (UD-O)	T-max (UD-C)	T-M (UD-O)	T-M (UD-C)
UD_Ancient_Greek	182030	12613	97	49	3.01	2.92
UD_Ancient_Greek-PROIEL	198034	15865	31	19	2.89	2.80
UD_Arabic	254120	6984	36	21	2.93	2.82
UD_Arabic-NYUAD	738889	19738	78	77	3.12	3.06
UD_Basque	97069	7194	13	13	2.25	2.20
UD_Belarusian	6864	333	17	9	2.48	2.29
UD_Bulgarian	140425	10022	14	10	2.24	2.18
UD_Catalan	474069	14832	20	15	2.69	2.61
UD_Chinese	111271	4497	27	17	3.24	3.18
UD_Coptic	8519	320	9	9	2.74	2.71
UD_Croatian	183816	8289	13	11	2.52	2.45
UD_Czech	1332566	77765	56	15	2.43	2.36
UD_Czech-CAC	483520	24081	47	12	2.50	2.37
UD_Czech-CLTT	26781	814	28	18	3.61	2.81
UD_Danish	90710	4947	16	16	2.61	2.58
UD_Dutch	197925	13050	15	15	2.89	2.86
UD_Dutch-LassySmall	91793	6841	29	13	2.74	2.56
UD_English	229733	14545	18	15	2.58	2.53
UD_English-LinES	67197	3650	25	11	2.54	2.43
UD_English-ParTUT	38114	1590	15	10	2.63	2.58
UD_Estonian	34628	3172	10	9	2.26	2.18
UD_Finnish	180911	13581	33	12	2.31	2.20
UD_Finnish-FTB	143326	16856	14	12	2.06	2.01
UD_French	392230	16031	34	11	2.51	2.41
UD_French-ParTUT	17927	620	11	11	2.70	2.64
UD_French-Sequoia	60574	2643	31	12	2.63	2.52
UD_Galician	109106	3139	15	15	2.56	2.56
UD_Galician-TreeGal	15436	600	13	10	2.55	2.44
UD_German	281974	14917	28	16	3.00	2.92
UD_Gothic	45138	4372	21	13	2.53	2.43
UD_Greek	51351	2065	13	11	2.51	2.45
UD_Hebrew	149088	5725	62	62	2.56	2.50
UD_Hindi	316274	14963	18	14	3.20	3.16
UD_Hungarian	31584	1351	13	12	2.83	2.77
UD_Indonesian	110143	5036	28	15	2.31	2.19
UD_Irish	13826	566	18	18	2.88	2.85
UD_Italian	282611	13402	35	14	2.50	2.41
UD_Italian-ParTUT	42651	1590	14	10	2.59	2.53
UD_Japanese	173458	7675	15	15	2.79	2.79
UD_Kazakh	529	31	8	7	2.67	2.61
UD_Korean	63426	5350	23	10	2.73	2.60
UD_Latin	18184	1334	17	11	2.86	2.80
UD_Latin-ITTB	280734	16508	11	11	2.67	2.65
UD_Latin-PROIEL	159407	15324	28	18	2.77	2.69
UD_Latvian	44795	3054	18	11	2.48	2.30
UD_Lithuanian	5356	263	14	10	2.43	2.31
UD_Norwegian-Bokmaal	280256	18106	38	12	2.44	2.38
UD_Norwegian-Nynorsk	276580	16064	38	12	2.50	2.43
UD_Old_Church_Slavonic	47532	5196	20	11	2.48	2.39
UD_Persian	136896	5397	14	14	3.45	3.44
UD_Polish	72763	7127	10	10	1.92	1.89
UD_Portuguese	217591	8891	19	13	2.54	2.47
UD_Portuguese-BR	287884	10874	38	11	2.54	2.45
UD_Romanian	202187	8795	14	11	2.39	2.33
UD_Russian	87841	4429	31	11	2.34	2.20
UD_Russian-SynTagRus	988460	55398	18	11	2.34	2.24
UD_Sanskrit	1206	190	8	6	2.23	2.19
UD_Slovak	93015	9543	10	10	2.00	1.97
UD_Slovenian	126593	7212	17	14	2.50	2.47
UD_Slovenian-SST	19488	2137	14	14	2.77	2.74
UD_Spanish	419587	15587	38	11	2.51	2.41
UD_Spanish-AnCora	496953	15959	31	15	2.63	2.58
UD_Swedish	76442	4807	31	27	2.58	2.49
UD_Swedish-LinES	64787	3650	25	11	2.55	2.43
UD_Tamil	9581	600	10	9	2.42	2.38
UD_Turkish	48093	4660	13	11	2.44	2.33
UD_Ukrainian	12846	863	11	9	2.19	2.11
UD_Urdu	123271	4595	32	16	3.44	3.36
UD_Uyghur	1662	100	8	8	2.93	2.92
UD_Vietnamese	31799	2200	8	8	2.09	2.06
Total	12101425	630518	97	77	2.62	2.55

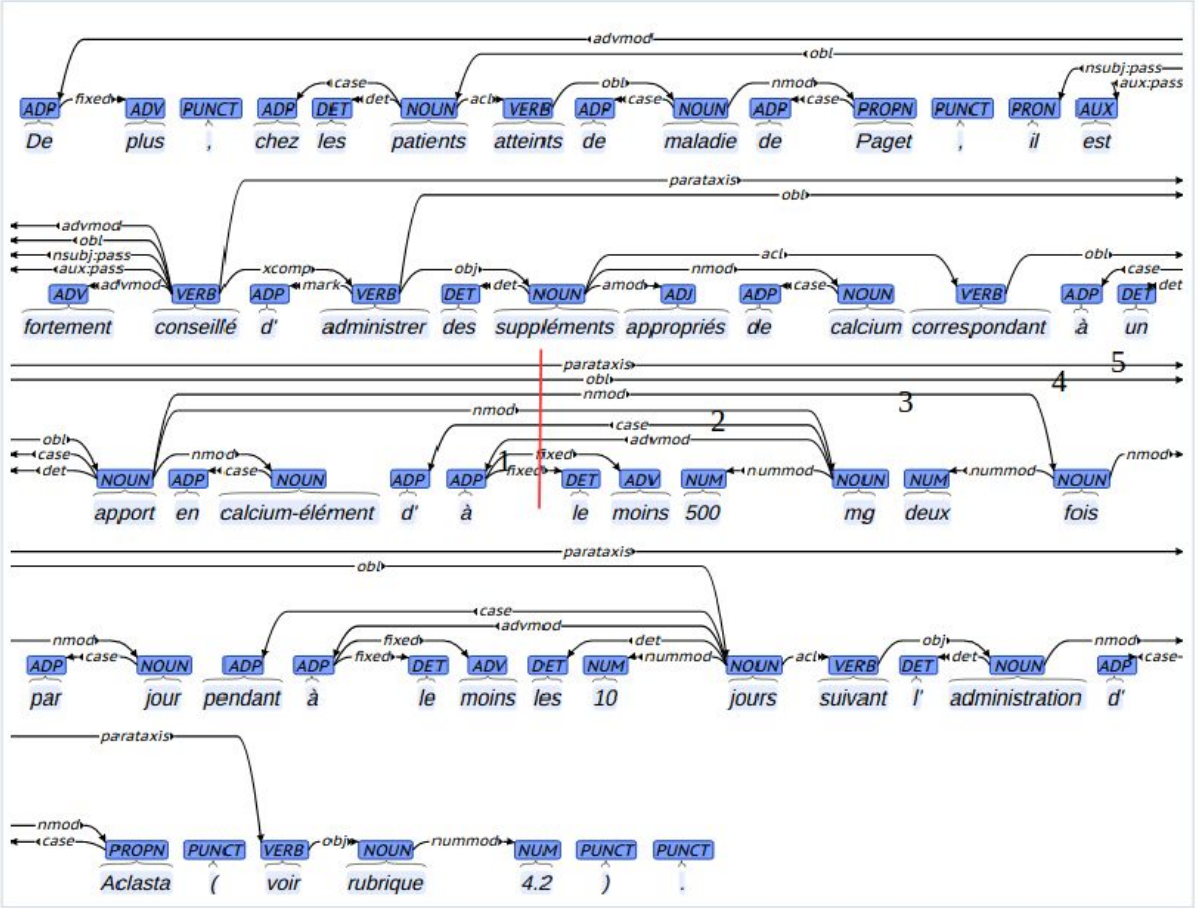
Tableau 6-3 Taille du flux pour UD-O (original) et UD-C (converti)

6.4 Exemples et constructions spécifiques

6.4.1 Les exemples ayant un poids de 5 ou 6

Les flux ayant un poids égal à 6 sont extrêmement rares. Il n'y a en a pas dans beaucoup de treebanks, notamment ceux pour le français. Dans cette partie, nous présentons les exemples de poids 5 pour la langue française et la langue anglaise, et 6 pour la langue chinoise du UD-original.

En français:



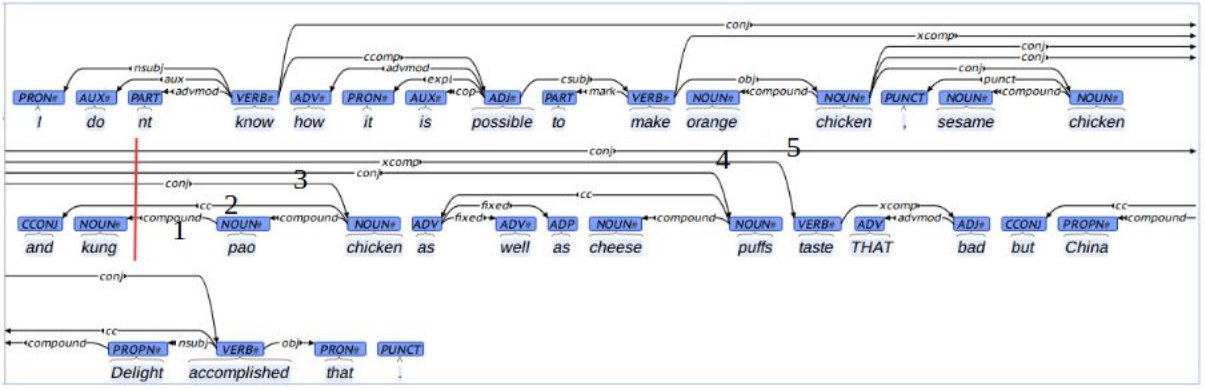
(fr_sequoia-ud-dev.conllu sent_id = 126)

"De plus, chez les patients atteints de maladie de Paget, il est fortement conseillé d' administrer des suppléments appropriés de calcium correspondant à un apport en calcium-élément d' à le moins 500 mg deux fois par jour pendant à le moins les 10 jours suivant l' administration d' Aclasta (voir rubrique 4.2) ."

Cette phrase contient un flux de poids 5 entre “à” et “le” [5], position que nous avons marquée par une ligne rouge.

1: fixed [à ADP , le DET]
   fixed [à ADP , moins ADV]
2: advmod [mg NOUN , d’ ADP]
   case [mg NOUN , à ADP]
   nmod [mg NOUN , apport NOUN]
3: nmod [apport NOUN , fois NOUN]
4: obl [administrer VERB , jour NOUN]
5: parataxis [conseillé VERB , voir VERB]

En anglais:



(en-ud-train.conllu sent_id = reviews-235423-0012)

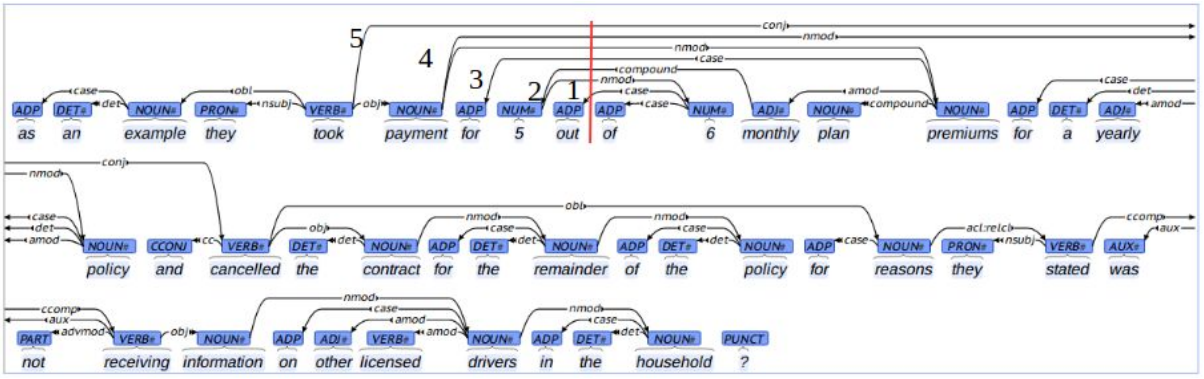
“I don’t know how it is possible to make orange chicken, sesame chicken and kung pao chicken as well as cheese puffs taste THAT bad but China Delight accomplished that.”

Cette phrase contient un flux de poids 5 entre “kung” et “pao”, position que nous avons marquée par une ligne.

1: compound [pao NOUN , kung NOUN]
2: cc [chicken NOUN , and CCONJ]
3: conj [chicken NOUN , chicken NOUN]
   conj [chicken NOUN , puffs NOUN]
4: xcomp [make VERB , taste VERB]

[5] Dans ce treebank, les articles contractés sont tokenisés en préposition et déterminant, ex: “aux” devient “à les”.

5: conj [know VERB , accomplished VERB]



(en-ud-train.conllu sent_id = reviews-217359-0006)

“as an example they took payment for 5 out of 6 monthly plan premiums for a yearly policy and cancelled the contract for the remainder of the policy for reasons they stated was not receiving information on other licensed drivers in the household ?”

Cette phrase contient deux flux de poids 5, nous montrons le flux entre “out” et “of” où nous avons tracé une ligne.

1: case [6 NUM , out ADP]

2: compound [monthly ADJ , 5 NUM]

nmod [5 NUM , 6 NUM]

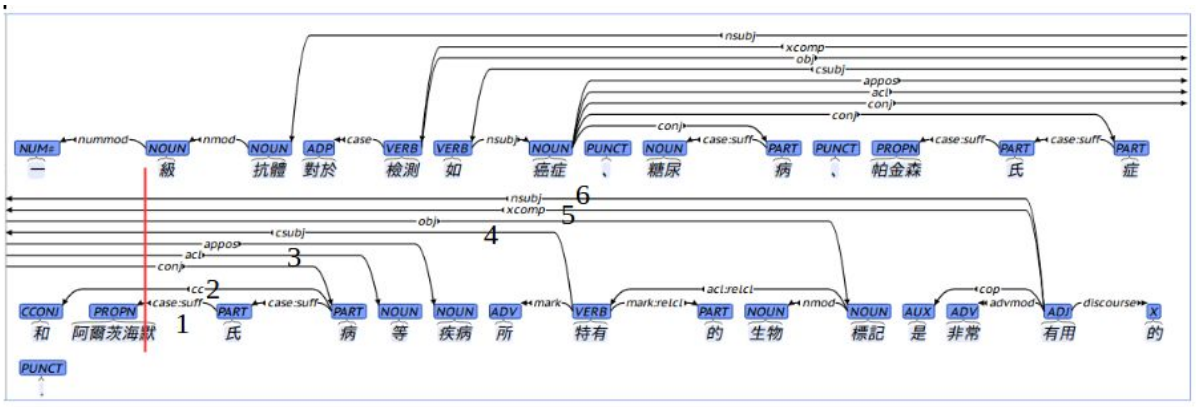
3: case [premiums NOUN , for ADP]

4: nmod [payment NOUN , premiums NOUN]

nmod [payement NOUN , policy NOUN]

5: conj [took VERB , cancelled VERB]

En chinois:



(zh-ud-train.conllu id=21)

一 級 抗體 對於 檢測 如 癌症 、糖尿 病 、 帕金森 氏

un niveau anticorps pour détecter comme cancer , diabètes maladie, Parkinson shi(suffixe)

症 和 阿爾茨海默 氏 病 等 疾病 所 特有

maladie et Alzheimer shi(suffixe) maladie etc. maladie qui contenir-spécifiquement

的 生物 標記 是 非常 有用 的 。

de(PART) biologie marqueur est très utile de(PART) .

Tr. Les anticorps primaires sont utiles pour détecter les biomarqueurs que les maladies telles que le cancer, le diabète, la maladie de Parkinson et la maladie d'Alzheimer, etc., contiennent spécifiquement.

Dans cet exemple, le flux du poids à 6 se trouve entre le nom ***阿爾茨海默*** *(Alzheimer)* et la particule ***氏*** *(shi (suffixe))* où l'on a tracé une ligne. Les numéros à côté représentent chaque niveau du poids :

1: case:suff [**氏** shi-*suffixe* PART , **阿爾茨海默** Alzheimer PROPN]
2: cc [**病** maladie PART , **和** et CCONJ]
3: conj [**癌症** cancer NOUN , **病** maladie PART]
acl [**癌症** cancer NOUN , **等** etc. NOUN]
appos [**癌症** cancer NOUN , **疾病** maladie NOUN]
4: csubj [**特有** contenir-spécifiquement VERB, **如** comme VERB]
5: obj [**檢測** détecter VERB, **疾病** maladie NOUN]
xcomp (**有用** utile ADJ, **檢測** détecter VERB)
6: nsubj (**有用** utile ADJ, **抗體** anticorps NOUN)

Nous trouvons que si nous le traduisons en français, le poids a beaucoup diminué, il est 3 au lieu de 6, en effet c'est l'ordre des mots qui provoque cette grande différence :

1. En chinois les adverbes, la négation, les compléments circonstanciels de temps ou de lieu, ainsi que les prépositions se placent normalement avant le verbe. Nous trouvons que dans la traduction "sont utiles" est plutôt au début de la phrase, pourtant dans la version chinoise , cet élément se trouve à la fin de la phrase, et le groupe prépositionnel "pour détecter les biomarqueurs" est avant le verbe principal.

2. Le modifieur de nom se place devant le nom. (‘[maladies [comme cancer, diabètes, Parkinson shi-*suffixe* maladie, et Alzheimer shi-*suffixe* maladie etc.]]’ devient ‘[[comme cancer, diabètes, Parkinson shi-*suffixe* maladie, et Alzheimer shi-*suffixe* maladie etc.] maladies]’)

3. Le chinois n'a pas de pronom relatif, et la proposition relative est avant le nom, ce qui pourrait provoquer complexité de la phrase par rapport aux autres langues SVO. Ce qui est également discuté dans le travaux de Hsiao F, Gibson E (2003) :

“*A key word-order difference between Chinese and other Subject-Verb-Object languages is that Chinese relative clauses precede their head nouns. Because of this word order difference, the results follow from a resource-based theory of sentence complexity, according to which there is a storage cost associated with predicting syntactic heads in order to form a grammatical sentence.*”

Par conséquent [**Les biomarqueurs** [(que) les maladies [comme le cancer, le diabète, la maladie de Parkinson et la maladie d'Alzheimer, etc.]] contiennent spécifiquement] devient [[[[comme le cancer, le diabète, la maladie de Parkinson et la maladie d'Alzheimer, etc.] les maladies] (qui) contiennent spécifiquement] **les biomarqueurs**].

6.4.2 Constructions spécifiques en chinois

D’après les résultats de UD-original, le treebank *UD_chinese* est un des corpus qui a le plus de poids par rapport aux autres langues (plus de 10% des positions ont un poids égal ou supérieur à 3), dans cette partie nous discutons des annotations qui pourraient augmenter le poids du flux.

La relation clf (ang. *classifier ;* fr. *classifieur*) pour la langue chinoise est utilisée pour indiquer le classifieur du nom.

En comparaison avec d'autres langues comme l'anglais ou le français dans lesquels un nom n'a qu'un déterminant, en chinois, un nom peut avoir son déterminant et son classificateur. Cette structure dans le corpus de *UD_chinese* est annotée comme: NOM -> CLF -> NUM. C'est une raison d’avoir plus de dépendances disjointes qui augmente le poids du flux.

Un exemple :

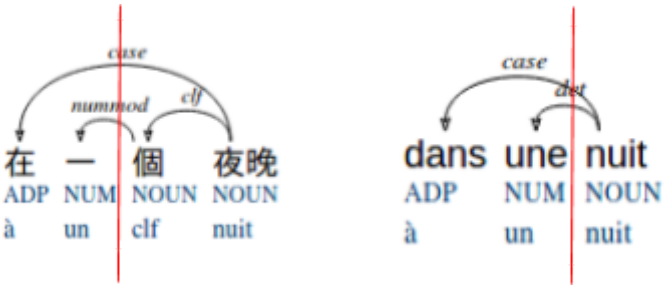


Figure 6-1

(Tiré dans *sent_id=1051* dans le fichier *zh-ud-train.conllu*)

在 一 個 夜晚

à un clf nuit

dans une nuit

Une autre analyse que nous proposons est de considérer le nom comme le gouverneur du classificateur et du déterminant comme la figure 6-2, dans le but de diminuer les dépendances disjointes:

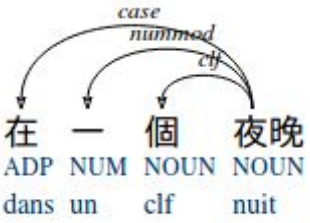


Figure 6-2

La relation *case:suff* est une relation entre le groupe nominal et son suffixe nominal. Comme l'exemple du chinois dans la section précédente (phrase *zh-ud-train.conllu id=21*), "Maladie d'Alzheimer" est annoté en chinois comme la figure 6-3 :

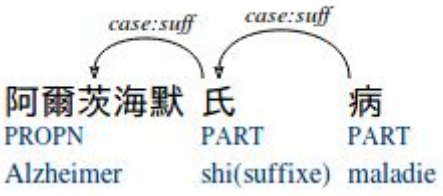


Figure 6-3

comparé avec l'annotation en français :

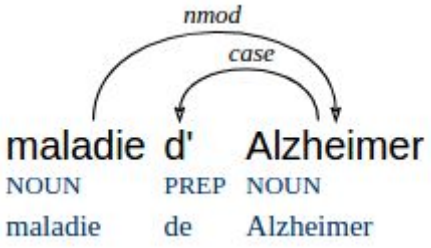


Figure 6-4

Mais la relation *case:suff* venant de l'analyse morphologique ne devrait pas être utilisée dans l'analyse syntaxique (voir Feng, 2000). Nous constatons que cette relation pourrait provoquer plus de dépendances disjointes par rapport au français, et que cela rend le poids plus lourd. D'ailleur, "maladie" 病 n'apporte pas d'information grammaticale, nous proposons NOUN au lieu de PART comme catégorie syntaxique. Finalement la relation syntaxique entre "maladie" 病 et "Alzheimer-shi*" 阿爾茨海默氏* qu'on propose serait *compound* dans la figure 6-4:

Figure 6-5

6.5 La comparaison de corpus oraux et écrits

Après avoir converti le corpus Rhapsodie en format UD[6], nous pouvons également faire une comparaison de corpus oraux et écrits. Rhapsodie (nous l'appelons *UD_Spoken-French*) est un corpus oral, et les trois autres venant de UD-original sont des corpus écrits (*UD_French, UD_French-ParTUT, et UD_French-Sequoia*).

[6] La conversion était faite automatiquement par logiciel OGRE (Optimized Graph Rewriting Engine (Ribeyre, 2013)) et script en Python.

	Tokens	Arbres	T-max	T-M	P-max	P-M
UD_French	392230	16031	34	2.51	5	1.39
UD_French-ParTUT	17927	620	11	2.70	5	1.44
UD_French-Sequoia	60574	2643	31	2.63	5	1.44
UD_Spoken-French	33551	2636	13	2.41	4	1.33

Tableau 6-4

D'après les résultats obtenus dans le tableau 6-4, la taille maximale du flux est de 13, et la taille moyenne est de 2.41, le poids maximal du flux pour *UD_Spoken-French* est de 4; Pour les trois autres il est de 5, le poids moyen est de 1.33. Nous pourrons remarquer que le poids maximal et le poids moyen dans *UD_Spoken-French* sont inférieurs aux trois autres corpus de l'écrit.

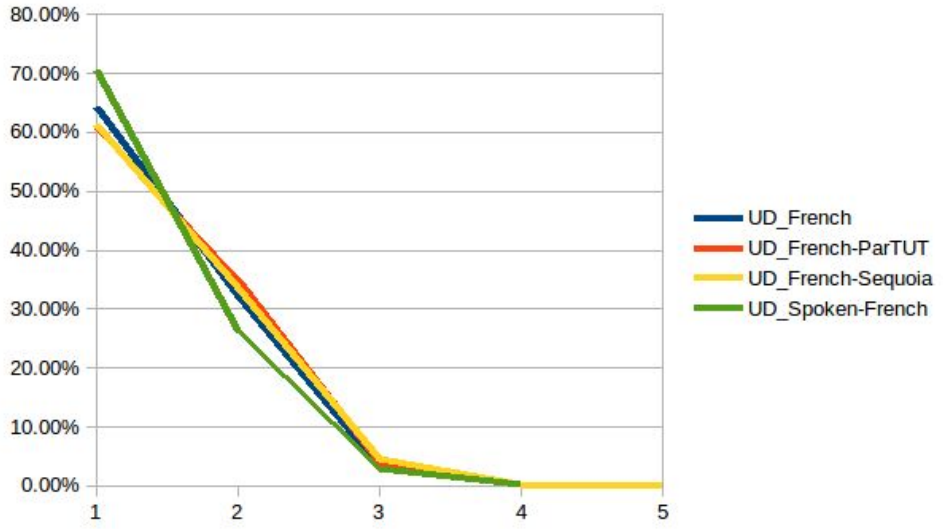


Figure 6-6

	Tokens	Arbres	1	2	3	4	5
UD_French	392230	16031	64.31%	32.23%	3.27%	0.17%	0.01%
UD_French-ParTUT	17927	620	60.66%	34.99%	4.05%	0.26%	0.03%
UD_French-Sequoia	60574	2643	61.25%	33.76%	4.69%	0.29%	0.01%
UD_Spoken-French	33551	2636	70.54%	26.35%	2.92%	0.19%	0.00%

Tableau 6-6

Si nous regardons en détail dans le tableau 6-6 illustré par la figure 6-6 , nous trouvons que les flux ayant un poids de 1 représentent 70.54% du total sur le corpus de français parlé, valeur la plus élevée parmi les quatre corpus de français. La limite du poids pour *UD_French, UD_French-ParTUT,* et *UD_French-Sequoia* est de 5, mais la limite du poids

pour *UD_Spoken-French* est de 4. Tous les corpus ont majoritairement des poids inférieurs ou égaux à 3.

Quant aux résultats sur l'empan gauche, l'empan droit, et la densité etc. Nous n'avons pas trouvé de corrélation entre les résultats et les différents genres de corpus (oral *vs* écrit).

Chapitre 7 Conclusion et perspectives

Nous avons étudié les propriétés du flux dans les 50 langues. En ce qui concerne la taille du flux, nous n'avons pas trouvé de limite universelle parmi ces langues, la représentation en chaîne pour la relation de la coordination ainsi que les relations similaires nous permettent de rendre le flux plus étroit et de diminuer la taille du flux.

Nos résultats sur l'empan et le rapport D/G pour les deux corpus UD-original et UD-converti ne se conforment pas idéalement à la distinction des langues à tête initiale et à tête finale. L'une des raisons pourrait être liée au schéma d'annotation de UD qui ne permet pas de prendre en compte la différence entre les langues à tête initiale et les langues à tête finale, ses annotations en bouquet pour les coordinations ainsi que les relations similaires pourraient rendre l'empan droit assez grand.

Le poids du flux (angl. *flux weight*) est la taille du plus grand sous-flux disjoint. D'après les résultats pour le poids du flux, nous avons trouvé que la limite est de 6 et qu'elle est très rarement atteinte (entre 0.05% et 0.00%). Cette limite de 6 pourrait être liée à la limitation de la mémoire immédiate.

Nous avons essayé d'analyser certaines particularités de l'annotation chinoise, comme la relation *clf* et *case:suff,* puisque dans le treebank de cette langue, plus de 10% des poids sont égaux ou supérieurs à 3 (alors que seuls 5.09 % des poids sont égaux ou supérieur à 3 sur l'ensemble des treebanks UD).

A la fin, en considérant que le genre du corpus pourrait influencer les résultats sur le flux, nous avons fait une comparaison entre corpus de l'écrit (*UD_French, UD_French-ParTUT, et UD_French-Sequoia*) et corpus de l'oral (*UD_Spoken-French* converti automatiquement à partir du corpus Rhapsodie). Les résultats montrent que le corpus *UD_Spoken-French* a un

poids moins lourd que les trois autres corpus de l'écrit : la valeur maximale est de 4 pour le premier et 5 pour les derniers, la valeur moyenne est de 2.41 contre 2.51, 2.70 et 2.63.

Pour aller plus loins, nous pouvons aller dans les trois directions ci-dessous. Premièrement, il nous faut convertir le treebank UD dans un schéma d'annotation plus convenable pour faire les calculs d'empan, afin de vérifier si nos résultats permettent de faire la distinction entre les langues à tête initiale et à tête finale; Deuxièmement, dans l'analyse de Lewis (1996), le niveau d'auto-enchâssement ne dépasse pas 3, nous envisageons de recalculer les poids pour une version de corpus sans considérer les relations des mots fonctionnels. Finalement, nous pourrons étudier précisément les divers genres de corpus français avec les propriétés du flux afin d'élaborer une typologie de genre pour le poids.

Bibliographies

Bloomfield, L. (1933). Language.

Botalla, M.-A. (2014). Analyse du flux de dépendance dans un corpus de français oral annoté en micro-syntaxe. Mémoire de master, Université Paris Ouest Nanterre La Défense

Candito, M., Crabbé, B., Denis, P., & Guérin, F. (2009, June). Analyse syntaxique du français: des constituants aux dépendances. In *16e Conférence sur le Traitement Automatique des Langues Naturelles-TALN 2009*.

Chomsky, N. (1957). Syntactic structures. The Hague, Mouton and Co.. 1965. Aspects of the theory of syntax.

Chomsky, N. (1965). Aspects of the Theory of Syntax.Cambridge MA: MIT Press

De Marneffe, M. C., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., & Manning, C. D. (2014, May). Universal Stanford dependencies: A cross-linguistic typology. In *LREC* (Vol. 14, pp. 4585-92).

Futrell, R., Mahowald, K., & Gibson, E. (2015). Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, *112*(33), 10336-10341

Gerdes, K., & Kahane, S. (2009). Speaking in piles: Paradigmatic annotation of french spoken corpus. In *Proceedings of the Fifth Corpus Linguistics Conference, Liverpool*.

Gerdes, K. (2013). Collaborative dependency annotation. *DepLing 2013*, *88*.

Gibson, E. (1998). Linguistic complexity: locality of syntactic dependencies. Cognition, 68, 1-76.

Hays, D. G. (1958, May). Grouping and dependency theories. In *Proceedings of the national symposium on machine translation*.

Hsiao, F., & Gibson, E. (2003). Processing relative clauses in Chinese. *Cognition*, *90*(1), 3-27.

Hudson, R. A. (1984). Word grammar. Oxford: Blackwell.

Jardonnet, U. (2009). Analyse du flux de dépendance. Mémoire de master, Université Paris Ouest Nanterre La Défense

Kahane, S. (2001). Grammaires de dépendance formelles et théorie Sens-Texte. *TALN 2001*

Kahane S. (avec la participation de K. Gerdes, P. Pietrandrea, C. Benzitoun, R. Bawden) (2012). Protocole de codage microsyntaxique. http://www.projet-rhapsodie.fr/

Kahane, S., & Mazziotta, N. (2015). Dependency-based analyses for function words Introducing the polygraphic approach. *Depling 2015*, 181.

Lacheret, A., Kahane, S., Beliao, J., Dister, A., Gerdes, K., Goldman, J. P., ... & Tchobanov, A. (2014, May). Rhapsodie: a prosodic-syntactic treebank for spoken french. In *Language Resources and Evaluation Conference*.

Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). Soar: An architecture for general intelligence. *Artificial intelligence*, *33*(1), 1-64.

Lewis, R. L. (1993). An architecturally-based theory of human sentence comprehension. In *Proceedings of the fifteenth annual conference of the cognitive science society* (pp. 108-113).

Lewis, R. L. (1996). Interference in short-term memory: The magical number two (or three) in sentence processing. *Journal of Psycholinguistic Research*, *25*(1), 93-115.

Liu, H. (2008). Dependency distance as a metric of language comprehension difficulty. *Journal of Cognitive Science*, *9*(2), 159-191.

Marcus, M. P. (1980). A theory of syntactic recognition for natural language. Cambridge MA: MIT Press

Marcus, M. P., Marcinkiewicz, M. A., & Santorini, B. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational linguistics*, *19*(2), 313-330.

Mel'čuk, I.A. (1988). Dependency syntax: Theory and practice. Albany: State University Press of New York.

Miller, G. (1956). The magical number seven plus or minus two: Some limits on our capacity for processing information. Psychological Review 63. 81–97.

Miller G. A, Chomsky, N. (1963). Finitary models of language users.

Murata, M., Uchimoto, K., Ma, Q., & Isahara, H. (2001, February). Magical number seven plus or minus two: Syntactic structure recognition in Japanese and English sentences. In *International Conference on Intelligent Text Processing and Computational Linguistics* (pp. 43-52). Springer Berlin Heidelberg.

Newell, A. (1994). Unified theories of cognition. Harvard University Press.

Nivre, J., Hall, J., & Nilsson, J. (2008). Memory-based dependency parsing.

Nivre, J., de Marneffe, M. C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., ... & Tsarfaty, R. (2016, May). Universal Dependencies v1: A Multilingual Treebank Collection. In *LREC*.

Petrov, S., Das, D., & McDonald, R. (2012). A universal part-of-speech tagset. In *Proceedings of LREC*.

Ribeyre, C. (2013, June). Vers un système générique de réécriture de graphes pour l'enrichissement de structures syntaxiques. I*n TALN 2013-20ème conférence du Traitement Automatique du Langage (pp. 178-191)*.

Rosenbloom, P. S., Laird, J., & Newell, A. (Eds.). (1993). The SOAR papers: Research on integrated intelligence.

Tesnière, L. (1959). Eléments de la syntaxe structurale. Paris: Klincksieck.

Wang, I., Kahane, S., & Tellier, I. (2014). Macrosyntactic Segmenters of a spoken French Corpus. In *9th Language Resources and Evaluation Conference (LREC)* (pp. 1-6).

Wehrli, E. (1989). Deux problèmes d'analyse syntaxique automatique. *Cahiers de Linguistique Française*, (10), 27-41

Yngve, V. H. (1960). A model and an hypothesis for language structure. *Proceedings of the American philosophical society*, *104*(5), 444-466.