Méthodes de représentation de la langue pour l'analyse syntaxique multilingue

Manon Scholivet
Aix-Marseille Univ, Université de Toulon, CNRS, LIS, Marseille
manon.scholivet@lis-lab.fr

RÉSUMÉ

L'existence de modèles universels pour décrire la syntaxe des langues a longtemps été débattue. L'apparition de ressources comme le *World Atlas of Language Structures* et les corpus des *Universal Dependencies* rend possible l'étude d'une grammaire universelle pour l'analyse syntaxique en dépendances. Notre travail se concentre sur l'étude de différentes représentations des langues dans des systèmes multilingues appris sur des corpus arborés de 37 langues. Nos tests d'analyse syntaxique montrent que représenter la langue dont est issu chaque mot permet d'obtenir de meilleurs résultats qu'en cas d'un apprentissage sur une simple concaténation des langues. En revanche, l'utilisation d'un vecteur pour représenter la langue ne permet pas une amélioration évidente des résultats dans le cas d'une langue n'ayant pas du tout de données d'apprentissage.

ABSTRACT

Language representation methods for multilingual syntactic parsing

The existence of universal models to describe the syntax of languages has been debated for decades. The availability of resources such as the World Atlas of Language Structures and the Universal Dependencies treebanks makes it possible to study the plausibility of universal grammar from the perspective of dependency parsing. Our work consists in studying different language representations in multilingual systems learned on treebanks in 37 languages. We show that representing the language of each word leads to better results compared to simply concatenate all the languages. However, using a vector to represent the language does not increase obviously the results in zero-shot learning cases.

MOTS-CLÉS : Analyse syntaxique multilingue, Représentation de la langue, Traits typologiques, WALS, UD.

KEYWORDS: Multilingual parsing, Language representations, Typological features, WALS, UD.

1 Introduction

L'analyse syntaxique en dépendances requiert, comme de nombreuses autres tâches en TAL, de grandes quantités de données pour apprendre la syntaxe de la langue. Certaines langues comme l'anglais bénéficient d'une grande quantité de données annotées. En revanche, pour de nombreuses langues, il n'en existe actuellement pas. L'analyse syntaxique multilingue est une solution possible pour tenter de tirer profit des langues bien dotées pour apprendre des informations communes entre les langues. Deux problèmes importants se posent alors : comment représenter les données, et comment représenter la langue ?

Avec l'apparition des Universal Dependencies (UD) [1], l'utilisation d'annotations universelles pour la syntaxe permet en partie de répondre au premier problème. En effet, UD propose un ensemble de relations de dépendances, de parties du discours et de traits morphologiques qui se veulent communs à toutes les langues (Nivre *et al.*, 2016). Les corpus arborés de UD sont disponibles pour de nombreuses langues, et sont basés sur un guide d'annotation commun.

Bien que les UD permettent un certain partage entre les langues au niveau des traits universels proposés, il n'existe cependant pas de lexique commun à toutes les langues. Le choix de l'utilisation d'un analyseur delexicalisé (Zeman & Resnik, 2008) a été fait pour ce travail. Cette technique consiste à ignorer le lexique lors de l'entraînement de l'analyseur. Cet appauvrissement des données conduit à des analyseurs moins précis, mais offre une solution simple au problème du lexique [2].

Le problème de représentation de la langue est le cœur de cet article. Trois méthodes de représentations seront présentées. En plus du simple identifiant de la langue, nous verrons comment tenter d'extraire les *paramètres* syntaxiques, premièrement à partir des données, ensuite en utilisant le World Atlas of Language Structures (WALS) [3] (Dryer & Haspelmath, 2013).

Nos contributions consistent à comparer ces différentes méthodes et à évaluer leurs effets dans un cadre d'entraînement multilingue, où le corpus d'entraînement est composé de corpus arboré de 37 langues, puis dans un cadre de *zero-shot*, c'est-à-dire pour une langue n'ayant pas de données d'entraînement (Ammar *et al.*, 2016a; Guo *et al.*, 2015).

Après la présentation de l'état de l'art (Sec. 2), nous décrirons les UD (Sec. 3), nos représentations des langues (Sec. 4), en particulier nous présenterons en détail le WALS (Sec. 4), puis présenterons l'analyseur utilisé (Sec. 5). Le cadre expérimental (Sec. 6) précédera nos résultats (Sec. 7) et nos conclusions (Sec. 8).

2 État de l'art

Ce travail est à l'intersection de trois tendances dans la littérature sur l'analyse syntaxique en dépendances multilingue. La première est le *transfer parsing*, qui consiste à apprendre un analyseur sur une langue (ou un ensemble de langues) puis à le tester sur une autre. La seconde est l' *analyse syntaxique delexicalisée*, qui a pour but de supprimer le lexique. La troisième et dernière tendance est l'utilisation de *ressources typologiques* telles que le WALS.

Le transfer parsing est une bonne solution lorsqu'il faut traiter des langues avec peu de ressources. McDonald *et al.* (2011) décrivent deux types de transfert : la première se base sur des corpus parallèles dont une des deux langues n'a pas, ou pas assez, de données d'entraînement pour apprendre un analyseur, alors que l'autre langue en a. La deuxième, les approches par transfert direct, reposent sur les similarités entre les langues et ne nécessitent pas de corpus parallèles. Par exemple, Lynn *et al.* (2014) proposent un analyseur pour l'irlandais entraîné d'abord sur une autre langue, puis appliqué à l'irlandais. Étonnamment, l'indonésien est la langue donnant les meilleurs résultats, bien qu'elles n'appartiennent pas à la même famille de langues. L'hypothèse des auteurs serait que les dépendances très distantes sont mieux représentées dans l'indonésien que dans les autres langues testées.

1. http://universaldependencies.org
2. Une alternative non explorée dans ce travail et que nous comptons explorer à l'avenir est l'utilisation de plongements de mots multilingues (Ammar *et al.*, 2016b)
3. https://wals.info/

Les langues pauvrement dotées peuvent parfois avoir une petite quantité de données d'entraînement disponible. En concaténant les corpus d'entraînement de deux langues, on peut obtenir un analyseur bilingue pour vérifier si une amélioration est possible comparé à un analyseur monolingue (Vilares *et al.*, 2015). Les méthodes de transfert direct et les analyseurs bilingues sont proches de cet article, puisqu'on retrouve cette idée de concaténation des corpus d'entraînement. Cependant, dans notre cas, nous combinons les corpus de bien plus de langues (environ 40) et incluons des méthodes de représentation des langues. La combinaison de corpus de multiples langues pour l'entraînement d'un analyseur est facilitée par les récentes avancées sur les standards multilingues et les ressources disponibles, en particulier grâce aux Universal Dependencies pour la syntaxe en dépendances (Nivre *et al.*, 2016). La recherche sur l'analyse syntaxique multilingue est fortement stimulée par des initiatives comme les campagnes d'évaluation CoNLL 2017 et 2018, sur l'analyse syntaxique en dépendances fortement multilingue à partir de textes bruts (Zeman *et al.*, 2017, 2018).

Les analyseurs delexicalisés ignorent la forme superficielle et les lemmes des mots lors de l'analyse d'une phrase, utilisant des traits plus abstrait comme les étiquettes de parties de discours (POS). L'utilisation d'analyseurs delexicalisés est particulièrement pertinente pour l'apprentissage d'analyseurs multilingues, puisque les langues partagent généralement peu de leurs lexiques. L'approche proposée par Zeman & Resnik (2008) consiste à adapter un analyseur pour une nouvelle langue en utilisant soit un corpus parallèle, soit un analyseur delexicalisé. Cette méthode peut être utilisée pour construire rapidement un analyseur si la langue source et la langue cible sont suffisamment proches.

De plus, les traits typologiques comme ceux présents dans le WALS donnent une information sur la structure des langues (Dryer & Haspelmath, 2013). Ces traits peuvent permettre à l'analyseur multilingue d'apprendre des caractéristiques communes à plusieurs langues. Naseem *et al.* (2012) et Zhang & Barzilay (2015) utilisent l'ensemble des traits issus de la catégorie de l'Ordre des mots du WALS qui sont disponibles pour leurs langues. Ponti *et al.* (2018) utilisent plutôt les traits qu'ils jugent pertinents dans les diverses catégories (pas uniquement celle concernant l'ordre des mots).

Täckström *et al.* (2013) utilisent une méthode de transfert multilingue delexicalisée, montrant en quoi le partage de paramètres basé sur des traits typologiques et l'appartenance à une famille de langues peut être utilisé dans un analyseur en dépendances discriminant. Les traits typologiques choisis sont basés sur ceux utilisés par Naseem *et al.* (2012), en retirant deux traits qu'ils ne considèrent pas utiles.

Les travaux les plus proches du notre concatènent des corpus pour entraîner un analyseur multilingue (Ammar *et al.*, 2016a). Les auteurs utilisent un analyseur à transitions S-LSTM similaire au notre (en dehors de l'utilisation de récurrence) entraîné sur un ensemble de traits lexicaux incluant des plongement de mots multilingues, des clusters de Brown, et des étiquettes POS [4] détaillées, alors que nous utilisons seulement des POS plus grossières et des traits morphologiques dans un cadre delexicalisé. Ils utilisent également un vecteur de représentation de la langue encodé en one-hot, l'ensemble de 6 traits issus du travail de Naseem *et al.* (2012), et testent en plus l'utilisation de la matrice complète du WALS. Nous avons testé ces deux premières représentations, ainsi qu'un autre vecteur de dimension 22 issu du WALS également. Leurs expériences ont été réalisées sur sept langues richement dotées alors que nous avons mené les nôtres sur un échantillon beaucoup plus grand de 40 langues. Bien que Ammar *et al.* (2016a) ont montré que, dans un cadre lexicalisé, la concaténation de corpus peut donner des résultats similaires à des analyseurs syntaxiques monolingues, les origines et les limites de ces gains restent floues. Nous explorons entre autres des pistes pour évaluer les avantages des caractéristiques typologiques dans un analyseur delexicalisé.

4. Partie du discours

3 Universal Dependencies

La cohérence de l'annotation des données entre les langues est un problème majeur sur les tâches d'analyse multilingue, la plupart des corpus étant annotés en utilisant différents guides d'annotation et jeux d'étiquettes. L'initiative des Universal Dependencies (UD) a pour but de créer des corpus arborés consistants entre les langues, facilitant ainsi les analyses lors de travaux sur la langue.

Nous utilisons la version 2.0 de UD pour nos corpus d'entraînement de développement [5], et les corpus de tests de la campagne d'évaluation CoNLL 2017 [6]. 64 corpus arborés de UD en 45 langues sont disponibles pour l'entraînement et le développement. Cependant, ces corpus sont de taille très variable (de 529 mots pour le kazakh à 1 1842 867 mots pour le tchèque). Les corpus de tests contiennent au minimum 10 000 mots par langue et sont disponibles pour 49 langues. 4 langues n'ont pas de corpus d'entraînement correspondant.

Les analyseurs delexicalisés, entraînés à partir des corpus de UD, prennent en entrée les parties du discours universelles (UPOS) et les traits morphologiques (FEAT) et prédisent les étiquettes de l'arbre de dépendances (incluant des sous-relations syntaxiques pouvant être spécifique au langage (p. ex., *acl :relcl*)). Les traits morphologiques sont renseignés pour presque tous les corpus, mais présentent de grandes différences. Par conséquent, nous avons fait le choix de ne garder que les traits les plus fréquents, qui apparaissent dans au moins 28 langues. De plus, les traits morphologiques sont représentés comme une liste de paires *(clef, valeur)* que nous avons séparé, afin que chaque paire soit considérée indépendamment, produisant ainsi un ensemble de 16 traits morphologiques par mot.

4 Les différentes représentations de la langue

Dans ce travail, nous utilisons trois méthodes de représentation de la langue. La première (que nous appelerons ID) consiste en un simple identifiant de la langue, sous la forme d'un vecteur one-hot de dimension 37 en entrée de notre analyseur. Les deux autres méthodes sont décrites ci-dessous.

Apprentissage à partir des données Une méthode pour représenter la langue consiste à apprendre un vecteur à partir de nos données d'entraînement, utilisant des statistiques sur les dépendances vues dans les corpus d'entraînement. Deux vecteurs seront appris de cette façon.

Le premier, que nous appellerons W_d, consiste à encoder, pour chaque étiquette des dépendances syntaxiques, la distance moyenne qui sépare les deux éléments de la relation dans une langue. Chaque dépendance d sera encodée sur deux composantes du vecteur W_d : une pour le cas où d est une dépendance droite, et l'autre pour les dépendances gauches.

La deuxième représentation, appelée W_{df}, ajoute à W_d l'information sur la fréquence de chacune de ces dépendances. Ainsi, si il existe n étiquettes de dépendance, W_d sera de taille $2 * n$ et W_{df} de taille $4 * n$. Un exemple de W_d et W_{df} est donné, dans la Table 1.

Ces vecteurs sont appris sur notre corpus d'entraînement (Sec. 6) et ne sont donc pas disponibles pour les langues n'ayant pas de corpus d'entraînement. Ils ne conviennent donc pas aux tests de type zero-shot, contrairement aux vecteurs issus du WALS présentés ci-dessous.

5. http ://hdl.handle.net/11234/1-1983
6. http ://hdl.handle.net/11234/1-2184

6,00	3,36	12,80	12,59	...
L_acl	R_acl	L_advcl	R_advcl	...

6,00	3,36	0,00005	0,02	12,80	...
L_acl	R_acl	freq_L_acl	freq_R_acl	L_advcl	...

TABLE 1 – Exemple de W_d (en haut) et W_{df} (en bas). L_acl représente les dépendances gauches étiquetées acl, dont la distance moyenne entre les éléments et de 6. R_acl représente la même chose, mais pour une dépendance droite. freq_L_acl est la fréquence des dépendances acl gauche dans la langue, et freq_R_acl est la fréquence des dépendances acl droite.

World Atlas of Language Structures Le World Atlas of Language Structures (WALS) est une base de données de propriétés structurelles (phonologiques, grammaticales et lexicales) réunie par 55 auteurs à partir de matériel descriptif comme par exemple des grammaires de référence. Cette base de données nous a permis d'associer à chaque langue des corpus de UD un ensemble de traits décrivant des propriétés pertinentes pour l'analyse syntaxique.

Le WALS décrit 2 676 langues grâce à un ensemble de 192 traits, répartis entre 11 familles (p.ex. Phonologie, Ordre des mots...). Il peut ainsi être représenté via une matrice W de 2 676 lignes et 192 colonnes, où chaque cellule $W(l, f)$ donne la valeur du trait f pour la langue l. Chaque ligne $W(l)$ est le vecteur de trait de la langue l.

Cette matrice a été épurée puis complétée pour correspondre à nos conditions expérimentales. Pour commencer, nous n'avons gardé que les lignes correspondant aux 49 langues de notre corpus de test. En revanche, 4 langues de UD (le Vieux Slave (cu), le Gothique (got), le Grec Ancien (grc), et le Latin (la)) n'apparaissent pas dans le WALS et ont donc été mises de côté. On obtient alors une version réduite de W contenant 45 lignes.

Deux de nos représentations de la langue ont été extraites à partir du WALS. La première, que l'on appellera W_N, est basée sur les travaux de Naseem *et al.* (2012), qui ont sélectionné les 6 traits [7] de la famille de l'Ordre des mots qui étaient entièrement renseignés pour leurs 17 langues cibles. Ces traits couvrent des phénomènes tels que l'ordre verbe-objet ou adjectif-nom, et ont été largement discutés dans la littérature (Täckström *et al.*, 2013; Zhang & Barzilay, 2015; Ammar *et al.*, 2016a). La matrice qui en résulte a 45 lignes (langues) pour 6 colonnes (traits). Cependant, le WALS est une matrice creuse, puisque certains traits ne sont pas renseignés pour certaines langues. C'est pourquoi nous avons fait le choix de ne garder que les langues pour lesquelles au plus la moitié du vecteur n'est pas renseignée, éliminant ainsi 5 langues de plus : le galicien (gl), le haut sorabe (hsb), le kazakh (kk), le slovaque (sk), et le ouïghour (ug). Nos expériences sont effectuées sur cet ensemble de 40 langues.

La deuxième représentation de la langue extraite du WALS, que l'on appellera W_{80}, est une version plus étendue de W_N. Nous nous sommes demandé s'il n'était pas dommage de se restreindre au traits de (Naseem *et al.*, 2012), qui ne gardent que 6 traits. Nous incluons alors dans W_{80} tous les traits renseignés pour au moins 80% de nos 40 langues.

En plus des traits de la famille de l'Ordre des mots, nous avons également inclus ceux de la famille des Propositions simples [8]. Il en résulte une matrice de 40 lignes et 22 colonnes, correspondant à 3 traits de la famille des Propositions simples (101A, 112A, 116A) et 19 traits de celle de l'Ordre des mots (81A, 82A, 83A, 85A, 86A, 87A, 88A, 89A, 90A, 92A, 94A, 95A, 96A, 97A, 144A, 143A, 143E, 143F, 143G).

7. Ces traits sont ceux identifiés par les codes 81A, 85A, 86A, 87A, 88A, 89A dans le WALS.
8. Nous avons également considéré la famille des Phrases Complexes, mais aucun trait ne dépassait le seuil des 80%.

	Romane	Germanique	Slave	Aléatoire
W_N	0.33	1.33	0.67	2.41
W_{80}	4.13	4.47	4.19	10.15

TABLE 2 – MID de chaque famille de langues comparée à l'aléatoire. Aléatoire est la moyenne de 50 000 familles de 6 langues.

Les matrices W_N et W_{80} obtenues ne sont cependant pas complètes : elles contiennent respectivement 4 et 35 valeurs non-renseignées, que nous avons rempli automatiquement. Chaque matrice W (abréviation pour W_N et W_{80}) permet de comparer deux langues l_1 et l_2 de manière simple en utilisant la distance de Hamming [9] entre leur vecteur $W(l_1)$ et $W(l_2)$, noté $d(l_1, l_2)$. Pour remplacer les valeurs manquantes, nous avons sélectionné, pour chaque langue l_1 contenant au moins une valeur non-renseignée (' ?'), la valeur correspondante dans le vecteur de la langue l_2 la plus proche qui soit entièrement renseignée, où $l_2 = \underset{l_i \mid \text{“?”} \notin W(l_i)}{\arg\min} d(l_1, l_i)$.

Les matrices W_N et W_{80} ne fournissent qu'une description partielle des langues, fortement biaisée en faveur de l'analyse syntaxique et ignorant les autres aspects (p.ex. la phonologie). Néanmoins, il est tentant de comparer les distances de langues appartenant aux mêmes familles typologiques dans ce mode de représentation. Pour cela, nous nous sommes intéressés à 3 familles présentes dans notre ensemble de 40 langues : les langues Romanes (6 langues), les langues Germaniques (6 langues) et les langues Slaves (7 langues). Nous avons alors calculé la proximité des vecteurs de ces langues. Nous définissons la distance interne moyenne (MID) d'un ensemble de langues $L = \{l_1, \ldots, l_n\}$, comme la moyenne des distances de chaque paire dans L :

$$MID(L) = \frac{1}{n^2 - n} \sum_{\substack{(l_i, l_j) \in L \times L \\ i \neq j}} d(l_i, l_j)$$

Nous avons calculé le MID de chaque famille de langues, et l'avons comparé au MID d'ensembles aléatoires de 6 langues (ce qui correspond au nombre de langues des familles Germanique et Romane). Les résultats de la Table 2 montrent clairement que les vecteurs du WALS permettent de capturer des similarités au sein d'une famille de langues, puisque le MID des vecteurs de langues d'une même famille est nettement inférieur au simple hasard.

5 Analyseur

L'analyseur utilisé dans nos expériences est un analyseur par transition de type *arc-eager* (Nivre, 2008), entraîné avec un oracle dynamique (Goldberg & Nivre, 2012). La prédiction des transitions est faite par un perceptron multi-couches (MLP) similaire au système de Chen & Manning (2014), consistant en une couche d'entrée, une couche cachée et une couche de sortie. Deux ensembles de traits entièrement delexicalisés ont été définis pour la prédiction : BASIQUE et ÉTENDU. BASIQUE est un ensemble classique composé de 9 traits liés au POS, 7 traits syntaxiques, 32 traits morphologiques et un trait de distance (la distance entre la tête et le dépendant). [10] L'ensemble ÉTENDU ajoute à BASIQUE de nouveaux traits correspondant aux vecteurs du WALS W_N et W_{80}, et/ou l'identifiant

9. Le nombre de dimensions pour lesquelles les valeurs diffèrent.
10. Nos corpus, le code source de l'analyseur, les fichiers de configuration et les matrices issues du WALS seront disponibles.

de la langue. Chaque trait est associé à un plongement de taille 3, initialisé à zéro. L'identifiant de la langue (issu de la représentation ID) est un vecteur one-hot de dimension 37 (correspondant aux 37 langues uniques). La couche d'entrée du MLP correspond à la concaténation des plongements des différents traits, dont les dimensions varient de 396 à 465 selon la configuration (avec ou sans vecteurs de langue W_N et W_{80}, ou de l'identifiant de la langue ID). La couche de sortie est composée de 263 neurones, correspondant au nombre de transitions que l'analyseur peut prédire. La couche cachée est de taille 1 000, avec un *dropout* durant l'entraînement de 0.4, le nombre d'itérations est égal à 10, la fonction d'activation est la fonction ReLu, la fonction objectif est un softmax de vraisemblance négative, et l'algorithme d'apprentissage est AMSgrad, utilisant les paramètres par défaut de Dynet (Neubig *et al.*, 2017). [11]

À chaque étape du processus d'analyse syntaxique, l'analyseur prédit une action à réaliser, pouvant aboutir à la création d'une nouvelle dépendance entre deux mots de la phrase. La prédiction des actions est basée sur la valeur des traits donnés au MLP. Dans la configuration BASIQUE, ces traits décrivent différents aspects du gouverneur, du dépendant, et du contexte. Par exemple, si la tête est un verbe et que le dépendant est un nom situé avant le verbe, une dépendance sujet aura une forte probabilité d'être prédite pour les langues utilisant majoritairement l'ordre sujet-verbe (SV). Avec l'ensemble de traits ÉTENDU, l'utilisation de l'ordre SV pour une langue est explicite. Le MLP a donc la possibilité de combiner une *configuration de phrastique* (p.ex., un nom avant un verbe) avec une *configuration de la langue* (p.ex., la langue est SV) quand il prédit une action. Les langues partageant un trait dans W auront la possibilité de générer la même prédiction pour une configuration phrastique correspondant à ce trait (p.ex. le nom précédant le verbe et la langue est de type SV). [12]

6 Cadre expérimental

Corpus Nos expériences ont été réalisées sur les données de la campagne d'évaluation CoNLL 2017 (Zeman *et al.*, 2017), en utilisant la tokenisation de référence et en ignorant les contractions (p.ex. *du=de+la*). Nos modèles sont évalués individuellement sur chacune des 40 langues pour lesquelles nous avons un vecteur $W(l)$ (Sec. 4), en utilisant les corpus de test de la campagne d'évaluation pour faciliter la comparaison avec les travaux similaires. Le corpus de test pour chaque langue est obtenu en faisant la concaténation de tous les corpus de test disponibles pour cette langue.

Trois langues n'ont pas de corpus d'entraînement ou de développement (bxr, kmr, sme). L'entraînement et le développement se font sur des corpus multilingues (ML) dérivés des 37 langues de UD restantes, que l'on nommera TRAIN-ML et DEV-ML. La taille des corpus de UD peut fortement varier d'une langue à l'autre, allant de 529 mots pour le Kazakh (kk) à 1 842 867 pour le Tchèque (cs). Ainsi, concaténer simplement tous les corpus pour constituer TRAIN-ML et DEV-ML sur-représenterait certaines langues et introduirait un biais en leur faveur. C'est pourquoi nous avons décidé d'équilibrer le nombre de tokens de TRAIN-ML et DEV-ML entre les langues.

La construction des corpus de développement et d'entraînement se fait en deux étapes. Tout d'abord

11. Les valeurs des hyperparamètres ont été définies dans des conditions similaires à $\Sigma\ W_N$, cf. Section 6).

12. Notre analyseur ne peut pas prédire d'arbre non-projectif. La présence d'une dépendence non-projective génère systématiquement une erreur d'analyse lors de la phase de test. Le taux moyen de projectivité du corpus de test est égal à 1%, avec un écart-type de 1% pour les 40 langues. 14 corpus (pour 20 langues) ont un taux de non-projectivité inférieur à 1%, et le taux maximum est de 8% pour le corpus du néerlandais (corpus Lassysmal). Nous avons fait des tests utilisant une transformation en arbre pseudo-projectif (Nivre & Nilsson, 2005), mais l'impact sur les résultats étant négligeable, nous avons choisi de garder l'algorithme projectif original.

la totalité des données des corpus disponibles (entraînement et développement) est divisée en deux à hauteur de 10% pour un sous corpus de développement et le reste pour l'entraînement. Ensuite, pour chacun des sous-corpus précédents, on sélectionne de façon aléatoire des phrases pour chaque langue jusqu'à ce que les corpus finaux atteignent une certaine limite en terme de tokens (respectivement 2 000 pour le DEV et 20 000 pour le TRAIN). Enfin les données ainsi sélectionnées sont mélangées (pour chacun des corpus) afin d'éviter de conserver l'ordre des langues. Avec cette procédure, la même phrase peut apparaître plusieurs fois. Néanmoins, cette approche garantit une représentation équilibrée de chaque langue dans TRAIN-ML et DEV-ML

Métrique La qualité des arbres prédits est évaluée par une mesure standard pour l'analyse syntaxique en dépendance : le score d'attachement de l'étiquette **AN étiqueté**(LAS). [13, 14] On donne le LAS par langue, ainsi que le MACRO-LAS, qui est la macro-moyenne du LAS de toutes les langues qui ont un corpus d'entraînement. Cette mesure est indépendante de la taille du corpus de test de chaque langue, et n'est pas biaisée en faveur des langues sur-représentées dans l'ensemble de test.

Configuration d'entraînement Nos expériences sur plusieurs paires ⟨corpus d'entraînement, représentation de la langue⟩ sont désignées par les codes suivants :

L : Corpus monolingue. Le corpus d'entraînement de la langue l consiste simplement à prendre toutes les phrases de la langue l dans TRAIN-ML. 37 analyseurs avec la configuration BASIQUE ont été entraînés, un pour chaque langue. Cette configuration correspond à la situation standard des expériences en analyse syntaxique : entraînement et test sur une même langue.

Σ : Corpus multilingue. Un analyseur est entraîné sur l'intégralité de TRAIN-ML, sans indication sur la langue. Le modèle est delexicalisé, le corpus ne contient donc que les étiquettes POS de référence, les traits morphologiques de référence, et les relations syntaxiques à apprendre.

Σ ID : Corpus multilingue + ID de la langue. Un analyseur entraîné avec la configuration ÉTENDU sur TRAIN-ML, en utilisant, attaché à chaque mot, l'identifiant de la langue.

$\Sigma\ W_N$, $\Sigma\ W_{80}$: Corpus + WALS. Deux analyseurs entraînés avec l'ensemble de traits ÉTENDU, sur TRAIN-ML. L'entraînement de $\Sigma\ W_N$ (resp. W_{80}) ajoute à chaque mot de Σ un vecteur $W(l)$ issu du WALS, qui correspond à la langue du mot. Un seul modèle est entraîné pour $\Sigma\ W_N$ (resp. W_{80}).

$\Sigma\ W_d$, $\Sigma\ W_{df}$: Corpus + vecteurs de données. Deux analyseurs entraînés avec l'ensemble de traits ÉTENDU, sur TRAIN-ML, avec W_d (respectivement W_{df}) attaché à chaque mot. Cet ajout est fait de la même façon que pour les tests $\Sigma\ W_N$ et $\Sigma\ W_{80}$.

$\overline{L}$ **:** *Zero-shot*. Dans cette configuration, 37 corpus d'entraînements sont dérivés de TRAIN-ML : pour chaque langue l, un corpus d'entraînement est construit à partir de toutes les phrases de TRAIN-ML, sauf celles appartenant à la langue l. Cette configuration représente la situation où un analyseur dans la configuration BASIQUE est entraîné pour une langue pour laquelle aucune donnée d'entraînement

13. Nous avons omis le score UAS car le UAS et le LAS sont étroitement corrélés ($r = 0.98$).
14. Nous avons utilisé le script d'évaluation de la campagne d'évaluation CoNLL 2017.

n'est disponible, comme dans les méthodes de transfert direct.

$\overline{L}\ W_{80}$ **:** $\overline{L}\ W_{80}$ ajoute à $\overline{L}$ l'indication de la langue de chaque mot à travers son vecteur du WALS, de la même manière que $\Sigma\ W_{80}$ faisait avec Σ.

7 Résultats et analyse

Nos expériences ont été réalisées dans les configurations décrites ci-dessus. Le LAS est donné pour chaque langue, ainsi que la macro-moyenne du LAS (MACRO), dans la Table 3. Nous commentons ci-dessous les résultats pour *L*, et comparons les résultats de certaines expériences (voir Table 4).

***L* :** Les résultats de l'expérience *L* montrent une importante variation des performances selon les langues. Le LAS varie de 46,78 pour le turc à 81,44 pour l'italien. Plus d'expériences seraient nécessaires pour expliquer les raisons d'une telle variabilité, mais n'entrent pas dans le cadre de ce travail. Nous pouvons tout de même émettre certaines hypothèses. Tout d'abord, certaines particularités spécifiques aux langues, comme l'équilibre entre les marqueurs morphologiques et syntaxiques (p.ex. les langues morphologiquement riches sont probablement favorisées dans notre configuration, puisque l'analyse morphologique est donnée en entrée de l'analyseur). D'autres sont spécifiques au genre textuel. Bien que la delexicalisation permette de neutraliser certains biais de genre, le genre peut aussi influencer la syntaxe, notamment la longueur des phrases (or, les phrases longues sont généralement plus dures à analyser), ou encore la proportion de constructions difficiles, comme les prépositions ou les coordinations ambiguës. Enfin, l'hétérogénéité de la qualité des annotations selon les langues peut également expliquer la variabilité du LAS.

***L* vs Σ :** Une chute attendue des performances est observée lorsque l'on passe de *L* à Σ. Le MACRO LAS chute de 5,68 points. L'hypothèse principale pour expliquer cette chute est le bruit qui est introduit par le mélange des langues. Concaténer toutes les phrases de plusieurs langues sans préciser l'identité de la langue introduit du bruit dans l'analyseur. Par exemple, la configuration phrastique associée à une dépendance sujet dans une langue SV ou VS sera très différente, et l'analyseur n'est pas capable de faire la distinction entre ces langues et voit donc des contradictions. La chute du LAS est cependant très variable selon les langues, allant même jusqu'à une augmentation plutôt qu'une chute dans le cas de l'espagnol (+0,51 points). Nous n'avons pas actuellement d'explication pour ce résultat, tout au plus une intuition qui serait que Σ apprendrait implicitement une langue moyenne (bruitée) qui serait plus proche de l'espagnol que du chinois par exemple (dont le LAS chute de 14,37 points), les langues composant Σ étant plus proches de l'espagnol que du chinois en moyenne.

Σ vs $\Sigma\ W_{80}$, $\Sigma\ W_d$, ID :[15] Nous obtenons ici le premier résultat important de ce travail : lors de l'ajout d'une représentation de la langue à l'analyseur, la MACRO LAS augmente de 4 à 4,75 points comparé à Σ. Le LAS augmente pour toutes les langues, pour les trois expériences. Ajouter l'information ID à Σ permet une augmentation des résultats MACRO de 4,60 points. Cette augmentation était attendue puisque dans ce test, les configurations phrastique sont associées à l'ID de la langue, ce qui aide à diminuer le bruit dans les données. Deux interprétations sont possibles pour ΣW_N : celle optimiste serait que les vecteurs de représentations de la langue aident à diminuer le bruit

15. W_N et W_{df} sont analysés plus tard.

L	Σ	Σ ID	$\Sigma\, W_N$	$\Sigma\, W_{80}$	$\Sigma\, W_d$	$\Sigma\, W_{df}$	$\overline{L}$	$\overline{L}\, W_{80}$	Lang.	
65,89	60,59	64,04	63,15	64,38	64,08	63,29	27,50	34,34	ar	
78,59	74,32	79,25	76,26	77,47	78,28	78,94	67,05	63,73	bg	S
77,18	72,76	76,63	73,03	76,27	76,90	77,17	70,48	68,88	ca	R
68,92	68,01	69,41	68,72	69,61	69,75	69,67	62,08	59,47	cs	S
73,62	67,38	70,26	70,19	70,25	70,57	72,19	61,56	63,87	da	G
71,07	63,76	70,36	69,18	69,22	70,37	70,44	59,65	60,51	de	G
77,11	71,26	76,16	73,29	75,84	76,11	76,90	63,74	65,96	el	
70,05	66,02	71,17	69,91	70,19	70,78	71,21	60,87	62,11	en	G
71,47	71,98	73,83	72,29	73,22	73,89	74,12	71,40	70,76	es	R
66,98	63,76	68,41	65,75	67,79	67,94	69,08	58,52	58,77	et	
63,26	55,76	60,11	60,22	59,39	60,00	60,31	33,50	31,68	eu	
72,85	66,02	69,50	69,63	70,00	69,67	68,02	31,25	34,27	fa	
60,97	56,29	59,65	57,37	59,28	59,99	59,92	50,69	48,72	fi	
75,74	74,25	76,16	74,79	75,82	75,80	76,50	72,68	71,97	fr	R
66,55	60,41	66,86	64,68	65,96	66,02	66,30	43,51	42,75	ga	
70,21	63,03	67,97	66,09	67,45	67,19	68,33	51,36	52,98	he	
78,91	73,86	74,86	75,77	74,45	76,66	76,52	54,23	57,33	hi	
71,03	67,49	71,37	70,00	70,40	71,42	71,28	62,88	65,71	hr	S
67,08	62,55	66,84	67,19	67,51	67,79	68,14	48,24	49,62	hu	
68,64	58,38	63,98	62,61	64,57	64,31	64,57	45,57	43,03	id	
81,44	76,45	80,56	76,97	79,83	80,45	81,73	75,82	77,04	it	R
78,26	68,22	75,81	74,85	75,56	76,59	76,89	7,64	28,53	ja	
47,68	37,17	38,61	38,07	39,66	41,11	39,36	17,99	23,18	ko	
59,89	54,11	59,98	58,23	60,17	59,78	60,86	44,41	47,38	lv	
62,56	57,21	59,28	58,13	58,59	59,11	60,00	51,11	48,98	nl	G
74,59	73,19	76,95	73,51	75,93	77,67	76,75	53,09	55,49	no	G
81,24	74,24	80,52	74,78	79,02	80,44	80,06	69,15	70,72	pl	S
72,00	65,74	69,83	68,74	69,86	69,88	69,96	62,85	65,86	pt	R
70,99	67,72	71,47	70,38	70,61	71,26	71,19	55,84	61,01	ro	R
74,06	61,35	74,72	68,65	74,45	74,92	75,06	55,09	55,50	ru	S
67,10	64,12	66,44	64,75	66,36	66,63	66,69	60,52	63,26	sl	S
72,05	69,97	72,55	70,71	71,88	73,21	73,25	64,80	67,05	sv	G
46,78	41,01	43,16	43,26	41,62	45,73	43,57	29,46	30,33	tr	
71,60	69,40	75,30	69,77	72,81	74,05	74,56	67,08	64,92	uk	S
74,35	69,15	70,93	70,71	70,76	72,41	71,47	58,93	59,41	ur	
54,40	42,42	53,75	51,94	51,72	52,81	52,42	25,86	41,07	vi	
59,83	45,46	58,48	53,42	54,87	56,95	59,19	22,24	24,48	zh	
69,32	63,64	68,25	66,41	67,64	68,39	68,54	51,86	53,80	MACRO	
-	33,32	34,10	30,37	28,49	-	-	-	-	bxr	
-	40,34	37,20	41,41	44,04	-	-	-	-	kmr	
-	47,34	45,60	47,63	42,38	-	-	-	-	sme	

TABLE 3 – LAS pour chaque langue, et MACRO LAS, pour les 9 configurations. Les langues suivies d'un S appartiennent à la famille des langues Slave, G appartiennent à la famille des langues Germaniques et R appartiennent à la famille des langues Romanes.

X	Y	$\overline{X-Y}$	σ	min	max
L	Σ	5.68	3.32	-0.51 es	14.37 zh
$\Sigma\ W_{80}$	Σ	4.00	2.58	0.59 hi	13.10 ru
Σ ID	Σ	4,60	2,91	1,00 hi	13,37 ru
$\Sigma\ W_d$	Σ	4,75	2,57	1,55 fr	13,57 ru
$\Sigma\ W_{80}$	$\Sigma\ W_N$	1.24	1.45	-1.64 tr	5.80 ru
Σ ID	$\Sigma\ W_{80}$	0,61	0,91	-1,05 ko	3,61 zh
$\Sigma\ W_d$	Σ ID	0,14	0,89	-1,53 zh	2,57 tr
$\Sigma\ W_{df}$	$\Sigma\ W_d$	0,15	0,89	-2,16 tr	2,24 zh
L	$\overline{L}$	17,47	13,57	0,07 es	70,62 ja
$\overline{L}\ W_{80}$	$\overline{L}$	1,95	4,60	-3,32 bg	20,89 ja

TABLE 4 – Différence entre les configurations X et Y : moyenne ($\overline{X-Y}$), écart type (σ), minimum et maximum avec la langue correspondante.

introduit par le mélange des langues dans Σ en "expliquant" certaines informations contradictoires dans les données grâce à l'utilisation des traits linguistiques encodés dans le WALS pour W_{80}, et grâce aux informations de distance contenues dans W_d. L'interprétation pessimiste consiste à dire que ces vecteurs sont un simple encodage arbitraire des langues. Dans ce cas, le MLP de l'analyseur apprendrait à associer les configurations phrastiques à certaines langues spécifiquement, et apprendrait alors différents modèles pour différentes langues, ce qui reviendrait à l'expérience ID. Plus d'expériences sont prévues pour comprendre comment le MLP utilise les vecteurs W.

$\Sigma\ W_N$ **vs** $\Sigma\ W_{80}$ **:** Les vecteurs W_N et W_{80} n'ont pas le même impact lorsqu'on les ajoute à Σ. L'ajout de W_{80} permet l'augmentation de 4 points de la mesure MACRO tandis que l'ajout de W_N augmente le score MACRO par rapport au MACRO obtenu avec Σ de 2,77 points seulement. L'analyseur est donc capable de tirer profit d'une description des langues plus riche lors de son apprentissage. Ce résultat pourrait indiquer que les résultats décevants sur l'analyse syntaxique rapportés par Ammar *et al.* (2016a), qui utilisaient le vecteur W_N, pourraient venir des traits extraits du WALS qui n'étaient pas suffisamment riches pour expliquer à l'analyseur les différences syntaxiques importantes entre les langues.

Σ ID **vs** $\Sigma\ W_{80}$ **:** Malgré de meilleurs résultats obtenus en utilisant W_{80} plutôt que W_N, ce nouveau vecteur n'est toujours pas capable de surpasser les performances de l'utilisation d'un simple identifiant de la langue. Mais l'augmentation de la mesure MACRO entre W_N et W_{80} peut laisser supposer que le choix des traits du WALS influence fortement les résultats. Plus d'expériences sont nécessaires pour confirmer cette hypothèse. On pourra également se poser la question de l'influence de la méthode de remplacement des valeurs inconnues.

Σ ID **vs** $\Sigma\ W_d$ **:** Le résultat de ces expériences représentent le deuxième résultat majeur de ce travail : le vecteur W_d permet de battre les résultats de Σ ID. Même si l'hypothèse qu'une partie du vecteur W_d sert a représenter la langue s'avère juste, une partie du modèle a été capable d'apprendre des informations supplémentaires, partagées entre les différentes langues. Ce résultat est tout de même à nuancer, puisque le LAS de $\Sigma\ W_d$ n'est supérieur que de 0,14 points à celui de Σ ID.

$\Sigma\ W_d$ **vs** $\Sigma\ W_{df}$ **:** Les résultats de $\Sigma\ W_d$ et de $\Sigma\ W_{df}$ sont assez proches (+0,15 points pour la mesure MACRO de W_{df}). L'information sur la fréquence des dépendances permet donc l'apprentissage de

connaissances supplémentaires pour l'analyseur. Cependant, le temps d'apprentissage de $\Sigma\, W_{df}$ étant assez conséquent, nous avons fait le choix de nous concentrer sur les résultats de $\Sigma\, W_d$. Le vecteur W_{df} étant deux fois plus gros que W_d, on peut également supposer que la taille de la couche cachée du MLP de l'analyseur n'est pas suffisante pour utiliser toutes les informations du vecteur W_{df}.

***L* vs $\overline{L}$:** $\overline{L}$ correspond à des conditions extrêmes mais également plus réalistes. La situation simulée est celle où aucune donnée d'entraînement n'est disponible pour la langue l. La chute des performances comparée à *L* était prévisible, mais n'en reste pas moins dramatique : la MACRO LAS chute de 17,47 points. Cette chute varie énormément selon les langues (cela se ressent sur l'écart type qui atteint 13,57 points). Certaines langues ne sont presque pas affectées, comme l'espagnol qui ne perd que 0,07 point. En revanche, le japonais perd 70,62 points de LAS, la plus grosse chute toutes expériences et langues confondues. Les langues les plus isolées sont celles qui souffrent le plus du passage à $\overline{L}$. Les familles de langues sont moins affectées, avec une chute de "seulement" 6,65 points en moyenne pour les langues romanes par exemple.

$\overline{L}$ vs $\overline{L}\, W_{80}$: Cette comparaison constitue notre troisième et dernier résultat majeur. Cette expérience consiste à observer si l'ajout d'un vecteur issu du WALS permet de limiter la chute des score LAS de l'expérience $\overline{L}$. Comme on peut le voir dans les Tables 3 et 4, l'ajout de W_{80} permet d'augmenter la MACRO LAS de 1,95 points, ce qui pourrait laisser penser que W_{80} permet bien d'aider le modèle à gérer les langues inconnues. Cependant, on remarque que l'écart type est assez élevé (4,60 points). En effet, lorsque l'on regarde le détail langue par langue, 11 de nos 37 langues ont de moins bons résultats avec le modèle $\overline{L}\, W_{80}$ qu'avec $\overline{L}$. Le bulgare perd même 3,32 points de LAS. Les langues ayant des résultats extrêmement bas dans $\overline{L}$ ont réussi à tirer profit du vecteur W_{80}, comme le japonais qui, après son importante chute au passage à $\overline{L}$, remonte de 20,89 points avec $\overline{L}\, W_{80}$. Le vecteur W_{80} reste une description extrêmement partielle de la langue, et il est probable qu'il ne soit pas suffisant pour compenser l'absence de corpus d'entraînement. La question précédemment soulevée se pose à nouveau : le vecteur sert il simplement d'identifiant de la langue ? Si l'analyseur utilise, ne serait-ce qu'en partie, W_{80} pour détecter la langue de la phrase analysée, l'absence de cette langue dans le corpus d'entraînement rend la tâche impossible.

8 Conclusions et travaux futurs

Dans ce travail, nous avons analysé des représentations possibles d'une langue dans le cadre d'une analyse en dépendances multilingue. Les meilleurs résultats sont obtenus avec les représentations apprises sur les données (W_d,W_{df}), bien qu'ils dépassaient de peu la représentation par un simple identifiant (ID). Mais ces représentations avaient l'inconvénient de ne pas être utilisables dans le cadre d'une langue pour laquelle il n'existe pas de données d'apprentissage, contrairement aux représentations utilisant le WALS. L'utilisation de ce vecteur ne permettant cependant pas une amélioration nette des résultats pour toutes les langues lors d'un d'apprentissage de type *zero-shot*.

Pour les travaux futurs, nous prévoyons de faire une analyse des poids du réseau afin de tester l'hypothèse selon laquelle W est utilisé comme un simple identifiant de la langue. Nous aimerions également tester d'autres méthodes de remplacement des valeurs non-renseignées du WALS, ainsi que la création d'un W_{de}, qui utiliserait l'écart type plutôt que la fréquence. Une autre étape sera l'introduction de plongements de mots multilingues pour donner des informations lexicales à l'analyseur.

Références

AMMAR W., MULCAIRE G., BALLESTEROS M., DYER C. & SMITH N. A. (2016a). Many Languages, One Parser. *arXiv :1602.01595 [cs]*. arXiv : 1602.01595.

AMMAR W., MULCAIRE G., TSVETKOV Y., LAMPLE G., DYER C. & SMITH N. A. (2016b). Massively Multilingual Word Embeddings. *arXiv :1602.01925 [cs]*. arXiv : 1602.01925.

CHEN D. & MANNING C. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 740–750 : Association for Computational Linguistics.

M. S. DRYER & M. HASPELMATH, Eds. (2013). *WALS Online*. Leipzig : Max Planck Institute for Evolutionary Anthropology.

GOLDBERG Y. & NIVRE J. (2012). A dynamic oracle for arc-eager dependency parsing. *Proceedings of COLING 2012*, p. 959–976.

GUO J., CHE W., YAROWSKY D., WANG H. & LIU T. (2015). Cross-lingual Dependency Parsing Based on Distributed Representations. In *ACL (1)*, p. 1234–1244.

LYNN T., FOSTER J., DRAS M. & TOUNSI L. (2014). Cross-lingual Transfer Parsing for Low-Resourced Languages : An Irish Case Study. In *Proceedings of the First Celtic Language Technology Workshop*, p. 41–49, Dublin, Ireland : Association for Computational Linguistics and Dublin City University.

MCDONALD R., PETROV S. & HALL K. (2011). Multi-source transfer of delexicalized dependency parsers. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, p. 62–72, Edinburgh, Scotland, UK. : Association for Computational Linguistics.

NASEEM T., BARZILAY R. & GLOBERSON A. (2012). Selective sharing for multilingual dependency parsing. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 629–637, Jeju Island, Korea : Association for Computational Linguistics.

NEUBIG G., DYER C., GOLDBERG Y., MATTHEWS A., AMMAR W., ANASTASOPOULOS A., BALLESTEROS M., CHIANG D., CLOTHIAUX D., COHN T., DUH K., FARUQUI M., GAN C., GARRETTE D., JI Y., KONG L., KUNCORO A., KUMAR G., MALAVIYA C., MICHEL P., ODA Y., RICHARDSON M., SAPHRA N., SWAYAMDIPTA S. & YIN P. (2017). Dynet : The dynamic neural network toolkit. *arXiv preprint arXiv :1701.03980*.

NIVRE J. (2008). Algorithms for deterministic incremental dependency parsing. *Computational Linguistics*, **34**(4), 513–553.

NIVRE J., DE MARNEFFE M.-C., GINTER F., GOLDBERG Y., HAJIC J., MANNING C. D., MCDONALD R., PETROV S., PYYSALO S., SILVEIRA N., TSARFATY R. & ZEMAN D. (2016). Universal dependencies v1 : A multilingual treebank collection. In N. C. C. CHAIR), K. CHOUKRI, T. DECLERCK, S. GOGGI, M. GROBELNIK, B. MAEGAARD, J. MARIANI, H. MAZO, A. MORENO, J. ODIJK & S. PIPERIDIS, Eds., *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France : European Language Resources Association (ELRA).

NIVRE J. & NILSSON J. (2005). Pseudo-projective dependency parsing. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, p. 99–106 : Association for Computational Linguistics.

PONTI E. M., REICHART R., KORHONEN A. & VULIĆ I. (2018). Isomorphic Transfer of Syntactic Structures in Cross-Lingual NLP. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 1531–1542, Melbourne, Australia : Association for Computational Linguistics.

TÄCKSTRÖM O., MCDONALD R. & NIVRE J. (2013). Target language adaptation of discriminative transfer parsers. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 1061–1071 : Association for Computational Linguistics.

VILARES D., GÓMEZ-RODRÍGUEZ C. & ALONSO M. A. (2015). One model, two languages : training bilingual parsers with harmonized treebanks. *arXiv :1507.08449 [cs]*. arXiv : 1507.08449.

ZEMAN D., POPEL M., STRAKA M., HAJIC J., NIVRE J., GINTER F., LUOTOLAHTI J., PYYSALO S., PETROV S. & POTTHAST M. (2017). CoNLL 2017 shared task : multilingual parsing from raw text to universal dependencies. *Proceedings of the CoNLL 2017 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies*, p. 1–19.

ZEMAN D., POPEL M., STRAKA M., HAJIC J., NIVRE J., GINTER F., LUOTOLAHTI J., PYYSALO S., PETROV S., POTTHAST M., TYERS F., BADMAEVA E., GOKIRMAK M., NEDOLUZHKO A., CINKOVA S., HAJIC JR. J., HLAVACOVA J., KETTNEROVÁ V., URESOVA Z., KANERVA J., OJALA S., MISSILÄ A., MANNING C. D., SCHUSTER S., REDDY S., TAJI D., HABASH N., LEUNG H., DE MARNEFFE M.-C., SANGUINETTI M., SIMI M., KANAYAMA H., DEPAIVA V., DROGANOVA K., MARTÍNEZ ALONSO H., ÇÖLTEKIN , SULUBACAK U., USZKOREIT H., MACKETANZ V., BURCHARDT A., HARRIS K., MARHEINECKE K., REHM G., KAYADELEN T., ATTIA M., ELKAHKY A., YU Z., PITLER E., LERTPRADIT S., MANDL M., KIRCHNER J., ALCALDE H. F., STRNADOVÁ J., BANERJEE E., MANURUNG R., STELLA A., SHIMADA A., KWAK S., MENDONCA G., LANDO T., NITISAROJ R. & LI J. (2018). CoNLL 2018 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies*, p. 1–19, Vancouver, Canada : Association for Computational Linguistics.

ZEMAN D. & RESNIK P. (2008). Cross-Language Parser Adaptation between Related Languages. In *Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages* : Association for Computational Linguistics.

ZHANG Y. & BARZILAY R. (2015). Hierarchical low-rank tensors for multilingual transfer parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, p. 1857–1867, Lisbon, Portugal : Association for Computational Linguistics.