Détection et correction automatique d'erreurs d'annotation morpho-syntaxique du French TreeBank

Florian Boudin Nicolas Hernandez
Université de Nantes
prénom.nom@univ-nantes.fr

RÉSUMÉ

La qualité de l'annotation morpho-syntaxique d'un corpus est déterminante pour l'entraînement et l'évaluation de méthodes d'étiquetage. Cet article présente une série d'expériences que nous avons menée sur la détection et la correction automatique des erreurs du *French Treebank*. Deux méthodes sont utilisées. La première consiste à identifier les mots sans étiquette et leur attribuer celle d'une forme correspondante observée dans le corpus. La seconde méthode utilise les variations de *n*-gramme pour détecter et corriger les anomalies d'annotation. L'évaluation des corrections apportées au corpus est réalisée de manière extrinsèque en comparant les scores de performance de différentes méthodes d'étiquetage morpho-syntaxique en fonction du niveau de correction. Les résultats montrent une amélioration significative de la précision et indiquent que la qualité du corpus peut être sensiblement améliorée par l'application de méthodes de correction automatique des erreurs d'annotation.

ABSTRACT

Detecting and correcting POS annotation in the French TreeBank

The quality of the Part-Of-Speech (POS) annotation in a corpus has a large impact on training and evaluating POS taggers. In this paper, we present a series of experiments that we have conducted on automatically detecting and correcting annotation errors in the French TreeBank. Two methods are used. The first simply relies on identifying tokens with missing tags and correct them by assigning the tag the same token observed in the corpus. The second method uses n-gram variations to detect and correct conflicting annotations. The evaluation of the automatic correction is performed extrinsically by comparing the performance of different POS taggers in relation to the level of correction. Results show a statistically significant improvement in precision and indicate that the POS annotation quality can be noticeably enhanced by using automatic correction methods.

MOTS-CLÉS : Étiquetage morpho-syntaxique, correction automatique, qualité d'annotation.

KEYWORDS: Part-Of-Speech tagging, automatic correction, annotation quality.

1 Introduction

Le corpus arboré de Paris 7, également appelé *French Treebank* (FTB), est la plus grande ressource disponible de textes annotés syntaxiquement et morpho-syntaxiquement pour le français (Abeillé

Actes de la conférence conjointe JEP-TALN-RECITAL 2012, volume 2: TALN, pages 281–291,
Grenoble, 4 au 8 juin 2012. ©2012 ATALA & AFCP

et al., 2003). Il est le résultat d'un projet d'annotation supervisée d'articles du journal Le Monde mené depuis plus d'une dizaine d'années. La quasi-totalité des méthodes d'étiquetage morpho-syntaxique du français utilisent cet ensemble de données que ce soit pour leur phase d'entraînement ou d'évaluation, e.g. (Crabbé et Candito, 2008; Denis et Sagot, 2010). La qualité de l'annotation du corpus est donc déterminante.

De la même manière que la plupart des corpus annotés morpho-syntaxiquement, e.g. le *Penn TreeBank* (Marcus *et al.*, 1993) pour l'anglais, le FTB a été construit de manière semi-automatique. Un étiqueteur automatique est d'abord appliqué sur l'ensemble des textes. Les sorties sont ensuite corrigées manuellement des éventuelles erreurs commises par l'outil. Malgré cette dernière étape, il est presque certain que des erreurs existantes ne sont pas corrigées et que de nouvelles erreurs sont introduites (les humains n'étant pas infaillibles). Plusieurs études illustrent d'ailleurs cette problématique en décrivant quelques unes des erreurs d'annotation récurrentes du FTB telles que l'absence d'étiquette ou la présence d'éléments XML vides [1] (Arun et Keller, 2005; Green *et al.*, 2011).

Dans cette étude, nous présentons une série d'expériences que nous avons menée sur la correction automatique du FTB. Nous détaillons les différentes erreurs que nous avons rencontrées ainsi que les solutions que nous appliquons. Deux méthodes sont utilisées. La première consiste à identifier les mots sans étiquette et leur attribuer celle d'une forme correspondante observée dans le corpus. La seconde méthode utilise les variations de *n*-gramme pour détecter et corriger les anomalies d'annotation. L'évaluation de la correction du corpus est réalisée de manière extrinsèque en étudiant l'impact du niveau de correction sur les performances de plusieurs méthodes d'étiquetage morpho-syntaxique.

Le reste de cet article est organisé comme suit. La section 2 présente le corpus *French Treebank* que nous utilisons dans cette étude. La section 3 est consacrée à la description de la méthode que nous proposons. Nous décrivons ensuite en section 4 nos résultats expérimentaux avant de présenter les travaux connexes aux nôtres. La section 6 conclut cette étude et donne quelques perspectives de travaux futurs.

2 Description du corpus *French Treebank*

Notre intérêt pour le FTB est motivé par deux objectifs : d'une part réaliser des traitements automatiques sur le corpus, et d'autre part, construire des modélisations permettant de prédire l'étiquette grammaticale d'un mot à l'aide d'approches statistiques ; ce deuxième objectif est un cas particulier du premier. Nos observations concernent donc à la fois la structure du corpus qui porte les annotations et la qualité de ses annotations grammaticales.

Le corpus est toujours en développement. La version que nous utilisons dans cette étude est datée de juillet 2010, elle est composée de 21 562 phrases pour 628 767 mots (*tokens*). Les fichiers qui composent le corpus sont au format XML (voir la Figure 1). Les mots sont répartis en 13 catégories principales (attribut `cat`), elles mêmes réparties en 34 sous-catégories (attribut `subcat`). De plus, les traits flexionnels (attribut `mph`), les lemmes (attribut `lemma`) et les mots composés

1. Certaines des erreurs recencées ne sont que des choix de représentation qui ne sont pas forcément des choix les plus adaptés dans une perspective de traitement automatique du corpus. C.f. `http://www.llf.cnrs.fr/Gens/Abeille/guide-morpho-synt.02.pdf` pour la représentation de « *du*» en deux balises tokens « *de*» et « *le*», la seconde ayant un contenu textuel vide.

(e.g. « *aujourd'hui* », « *mettre en garde* ») sont explicités. Ces derniers sont très nombreux dans le corpus : ≈14% des occurrences de tokens entrent dans un mot composé. On peut noter que la structure originale de la phrase (avec les caractères espaces) ainsi que l'identifiant du document source ne figurent pas dans le corpus.

```
<SENT nb="226">
 <NP>
  <w cat="D" [...] lemma="son" mph="1fss" subcat="poss">Ma</w>
  <w cat="N" [...] lemma="position" mph="fs" subcat="C">position</w>
 </NP>
 <VN>
  <w cat="V" [...] lemma="être" mph="P3s" subcat="">est</w>
 </VN>
 <NP>
  <w cat="D" [...] lemma="le" mph="fs" subcat="def">la</w>
  <w cat="N" [...] lemma="suivant" mph="fs" subcat="C">suivante</w>
 </NP>
 <w cat="PONCT" [...] lemma="." subcat="S">.</w>
</SENT>
```

FIGURE 1 – Exemple de phrase annotée extraite du fichier `lmf300_13000ep.cat.xml`, certains attributs ont été supprimées pour faciliter la lecture (`[...]`).

L'encodage natif du corpus est *iso-8859-1*. Le premier traitement que nous avons opéré est sa conversion en *utf-8* via l'outil GNU iconv. L'encodage *utf-8* est utilisé par défaut par l'ensemble des outils et des applications que nous utilisons. Seul le fichier `lmf7ad1co.aa.xml` fut récalcitrant et nous avons été amené à corriger les caractères accentués à l'aide de quelques règles de conversion ad hoc. Le FTB nécessite ensuite de nombreux pré-traitements avant de pouvoir être utilisé pour l'entraînement et l'évaluation d'étiqueteurs morpho-syntaxiques. Le format XML d'origine doit tout d'abord être converti au format d'entrée standard [2]. Cette première conversion du corpus nous a permis d'identifier et de corriger quelques problèmes liés à sa structure : absence d'attribut, étiquette de catégorie morpho-syntaxique non valide, etc.

2.1 Choix du jeu d'étiquettes et du découpage en unités lexicales

Plusieurs possibilités s'offrent à nous quant au choix du jeu d'étiquettes morpho-syntaxiques. (Crabbé et Candito, 2008) ont proposé un jeu d'étiquettes optimisé en 29 catégories (utilisant l'information supplémentaire du mode des verbes et de certaines sous-catégories). Les résultats obtenus avec leur méthode indiquent une amélioration de la précision par rapport à l'utilisation du jeu de 13 étiquettes du FTB. Cependant, les tokens présents dans les mots composés ne contiennent que l'information de la catégorie principale (attribut `catint`). Il n'est donc pas toujours possible de leur attribuer une étiquette optimisée automatiquement. La solution retenue par (Arun et Keller, 2005) et les travaux suivants consiste à fusionner les tokens et de leur affecter l'étiquette du mot composé. Par exemple, les tokens du mot composé illustré dans la Figure 2 seront fusionnés en « `levée_de_boucliers` » avec l'étiquette `NC (cat="N"+subcat="C")`. Cette méthodologie simplifie artificiellement la tâche d'étiquetage mais facilite la comparaison avec les approches précédentes.

2. Une phrase par ligne dans laquelle chaque mot est suivi d'un séparateur et de son étiquette. Par exemple, la phrase de la Figure 1 doit être converti en : `Ma/D position/N est/V la/D suivante/N ./PONCT`

```
<w cat="N" [...] lemma="levée de boucliers" mph="fs" subcat="C">
  <w catint="N">levée</w>
  <w catint="P">de</w>
  <w catint="N">boucliers</w>
</w>
```

FIGURE 2 – Exemple de mot composé extrait du fichier `lmf300_13000ep.cat.xml`.

Néanmoins, entraîner des méthodes d'étiquetage avec un ensemble de données dans lequel les mots composés sont fusionnés suppose par la suite l'utilisation d'un *tokenizer* capable de détecter les mots composés. Or, les méthodes existantes ne sont pas encore arrivées à un niveau de maturité satisfaisant. De plus, la notion de mot composé reste encore ambiguë et spécifique aux choix faits par les annotateurs du FTB. En effet, la définition du mot composé dans le FTB est assez large, avec par exemple « *à tout prix* » ou « *seconde guerre mondiale* ». Nous avons fait ici un choix restrictif en ne fusionnant que les mots composés lexicaux dont le lemme ne contient pas le caractère espace (e.g. « *aujourd'* » + « *hui* » → « *aujourd'hui* », « *celles* » + « - » + « *ci* » → « *celles-ci* ») ainsi que les nombres décimaux (e.g. « *16* » + « *,* » + « *7* » → « *16,7* ») et découpés (e.g. « *500* » + « *000* » → « *500000* »). Un total de 8 967 mots-composés sont fusionnés de cette manière.

Pour l'ensemble des raisons que nous avons évoquées précédemment, nous utilisons dans cette étude le jeu de 13 étiquettes dérivées des catégories principales du FTB. Ce choix est également appuyé par le fait que nous souhaitons proposer un ensemble de règles de corrections automatiques ne nécessitant pas de ressources externes ou d'intervention manuelle. Le corpus généré à partir de cette conversion directe du FTB contient 7 747 tokens pour lesquels aucune étiquette n'a pu être affectée, i.e. soit l'étiquette est manquante, soit l'étiquette présente n'est pas valide. Au total, le corpus généré contient 2 090 phrases dans lesquelles au moins un token sans étiquette est présent.

D'un point de vue pratique, la plupart des systèmes d'étiquetage nécessitent des données d'entraînement complètement étiquetées (i.e. sans étiquette manquante). Nous attribuons donc l'étiquette U (pour *Unknown*) aux tokens pour lesquels aucune étiquette n'a pu être affectée.

3 Correction automatique des erreurs d'annotation

La méthodologie de correction automatique des erreurs d'annotation peut être décomposée en deux étapes : i. identifier les occurrences des mots incorrectement étiquetés (ou ayant une étiquette manquante) dans le corpus ; ii. assigner les bonnes étiquettes correspondant à ces occurrences. Concernant la seconde étape, nous avons avant tout cherché à privilégier la précision des corrections. Ainsi, notre choix s'est porté sur des méthodes cherchant à assigner une étiquette corrective avec la plus grande confiance possible au détriment du nombre d'erreurs candidates corrigées.

Nous proposons deux méthodes pour corriger les erreurs : la première vise la correction des étiquettes manquantes de certains mots à l'aide de la fréquence d'occurrence des étiquettes associées à d'autres occurrences du mot (Section 3.1), que nous désignerons par FTB+corr. 1. La seconde vise la correction des erreurs d'annotation par la détection des variations d'étiquetage pour des *n*-grammes de mots (Section 3.2), que nous désignerons par FTB+corr. 2.

3.1 Correction des étiquettes manquantes

Une solution simple au problème des étiquettes manquantes consiste à attribuer l'étiquette de la forme correspondante dans le corpus. Dans le cas ou plusieurs étiquettes ont été attribuées à un même token, la plus fréquente sera choisie. Cette stratégie peut s'avérer être problématique dans le cas où les fréquences des différentes étiquettes d'un token sont égales ou très proches. Par exemple, « *quelque* » apparaît 47 fois en tant qu'adverbe, 46 fois en tant que déterminant et 34 fois en tant qu'adjectif. Le choix de l'étiquette serait dans ce cas ambigu. Pour minimiser le risque d'introduire des erreurs d'annotations, nous n'attribuons l'étiquette la plus fréquente que si sa fréquence dans le corpus est supérieure à la somme des fréquences des autres étiquettes candidates. Seules les étiquettes avec une fréquence supérieure à 1 sont utilisées.

Le nombre de tokens sans étiquette est ainsi ramené à 901, tandis que le nombre de phrases contenant au moins une étiquette manquante est réduit de 2 090 à 582. Malgré les contraintes que nous avons mises en place, il est probable que cette méthode de correction introduit des étiquettes erronées. La seconde méthode que nous décrivons dans la section suivante permet de détecter et de corriger les éventuelles séquences d'étiquettes erronées.

3.2 Détection et correction des variations d'annotation

Afin de détecter les erreurs d'annotation nous mettons en oeuvre l'approche proposée par (Dickinson et Meurers, 2003) puis reprise par (Loftsson, 2009) pour évaluer les corpus *Wall Street Journal* (WSJ) et *Icelandic Frequency Dictionary* (IFD). L'approche repose sur la détection de variations d'étiquetage pour un même *n*-gramme de mots. On utilisera le terme de *variation de n-gramme* (*variation n-gram*) pour désigner un *n*-gramme de mots dans un corpus qui contient un mot annoté différemment dans une autre occurrence du même *n*-gramme dans le corpus. Le(s) mot(s) sujet(s) à la variation (qui ont une étiquette différente dans les différentes occurrences) est(sont) appelé(s) *noyau de variation* (*variation nucleus*).

La présence au sein d'un corpus d'une variation d'annotations pour un même *n*-gramme de mots peut s'expliquer soit par l'ambiguïté des mots noyaux de la variation (une même forme peut admettre des étiquettes distinctes dans un contexte d'occurrence différent) soit par une erreur. L'hypothèse que l'on pose est : plus des contextes d'une variation sont similaires, plus grande est la probabilité qu'il s'agisse d'une erreur. La notion de contexte se traduit ici par le nombre de mots, *n*, que l'on considère dans les *n*-grammes observés. La table 1 rapporte une comparaison en chiffres des observations menées sur les différents corpus traités par cette méthode.

Comme l'explique (Loftsson, 2009), une même erreur candidate peut être détectée plusieurs fois du fait du fait qu'un *n*-gramme de mots peut se retrouver contenu dans un autre pour une valeur de *n* supérieure. De plus, une variation de *n*-gramme contient à minima deux annotations possibles pour le même *n*-gramme de mots. Il n'est ainsi pas facile de calculer la précision de cette méthode (i.e. le ratio d'erreurs correctement détectées sur toutes les erreurs candidates).

(Dickinson et Meurers, 2003; Loftsson, 2009) avaient pour objectif d'évaluer manuellement le nombre de variations distinctes correctes. Pour cette raison, ils ont choisi un *n* minimal suffisamment grand pour que le contexte soit discriminant. Ils ont par ailleurs considéré la plus longue variation de *n*-grammes pour chaque occurrence de mot présentant une variation afin d'avoir le plus de matériel sous les yeux pour permettre la levée de l'ambiguïté. Notre objectif

Corpus	WSJ	IDP	FTB+corr. 2	FTB+corr. 1&2
# tokens	1,3 M	590 297	628 767	
# étiquettes	36	700	13**	
+ longue variation	224	20	87	
Valeur de n observé	$n \geq 6$	$n \geq 5$	$n \geq 5$	
Variations distinctes	2495	752	293	147
Vraies erreurs	2436 (97,6%)	254	–	–
# tokens corrigés	4417 (0,34%)	236* (0,04%)	741	169

TABLE 1 – Comparatifs des corpus en chiffres sur lesquels des *variations de n-gramme* ont été calculées. * Nous constatons que le nombre réel de tokens corrigés, calculé à partir du pourcentage fourni par (Loftsson, 2009), est inférieur au nombre de variations étant de vraies erreurs ; nous supposons que cela est peut être dû à une erreur dans le recensement des variations distinctes observées. ** A ce nombre il faut rajouter une étiquette supplémentaire que l'on utilise pour tous les mots qui n'ont pas nativement une des 13 étiquettes retenues.

diffère puisque nous souhaitons détecter et corriger des erreurs automatiquement. Nous sommes néanmoins sensibles au fait de poser un n suffisamment grand pour discriminer mais aussi suffisamment petit pour que la différence du nombre d'occurrences entre les variations puisse être utilisée pour filtrer les variations les moins probables. Du fait que l'IDP et le FTB ont une taille proche, nous suivons le choix de (Loftsson, 2009) et optons pour $n \geq 5$.

Nous proposons une heuristique pour corriger certaines variations. Celle-ci est la suivante : nous considérons les n-grammes par taille décroissante, puis par nombre d'occurrences décroissant. Nous sélectionnons les candidats pour une correction selon deux contraintes : i. la présence d'au moins deux unités lexicales et ii. la présence d'une variation, sans étiquette manquante, dont le nombre d'occurrence est strictement supérieur à la somme des occurrences des autres variations. De fait seuls les n-grammes apparaissant au moins trois fois sont considérés. Cette dernière contrainte nous sert aussi de base pour proposer une correction. En effet, la variation qui valide la contrainte est considérée comme la séquence d'étiquettes correcte.

Les exemples 1, 2 et 3 illustrent des corrections opérées avec cette heuristique. Les mots corrigés sont soulignés.

(1) ,/PONCT l'/D une/N des/P plus/ADV → ,/PONCT l'/D une/PRO des/P plus/ADV

(2) produit/N intérieur/N brut/A (/PONCT PIB/N)/PONCT → produit/N intérieur/A brut/A (/PONCT PIB/N)/PONCT

(3) d'/P état/N chargé/N de/P la/D → d'/P état/N chargé/V de/P la/D

Pour $n \geq 5$, lorsque l'on applique cette méthode directement sur le FTB, nous comptons 293 variations distinctes (vérifiant les contraintes citées ci-dessus) et le nombre de tokens corrigé est 741 (dont 593 étaient sans étiquette). Le nombre de tokens corrigés augmente lorsque l'on diminue la taille minimale des n-grammes traités. Nous avons néanmoins préféré garder un n suffisamment haut pour maintenir une certaine confiance dans le choix de considérer certaines des variations détectées comme erreurs.

Intrinsèquement la méthode par détection de variations de n-gramme repose sur le nombre d'occurrences des n-grammes. La méthode est donc sensible à la taille du corpus et l'on peut

s'attendre à ce qu'elle fournisse de meilleurs résultats sur des corpus homogènes (i.e. d'un genre spécifique) utilisant un jeu d'étiquette à gros grain ; caractéristiques que nous retrouvons dans le FTB.

4 Résultats

L'évaluation des corrections apportées au corpus est réalisée de manière extrinsèque. L'idée derrière cette méthodologie est simple, il s'agit de comparer les scores de performance de différentes méthodes d'étiquetage morpho-syntaxique en fonction du niveau de correction du FTB. Une amélioration de la précision de l'étiquetage est une indication indirecte de la bonne correction du corpus.

Les méthodes d'étiquetage morpho-syntaxique fondées sur des modèles probabilistes discriminants atteignent des niveaux de performance très élevés. Dans cette étude, nous avons choisi deux systèmes utilisant des modèles par maximum d'entropie (*MaxEnt*) : la version 3.0.4 du *Stanford POS Tagger* (Toutanova *et al.*, 2003) et l'étiqueteur morpho-syntaxique de la suite *Apache OpenNLP* [3]. Le *Stanford POS Tagger* a été entraîné avec un ensemble standard [4] de traits bidirectionnels sur les mots et les étiquettes. L'étiqueteur d'*Apache OpenNLP* a, quant à lui, été entraîné avec l'implémentation par défaut qui caractérise chaque mot à l'aide de traits caractéristiques des trois mots précédents et suivants. Ces traits sont les préfixes et les suffixes de quatre caractères, la classe rudimentaire d'information de ces caractères (e.g. débute avec une majuscule, est un nombre, est un symbole), l'étiquette grammaticale et la forme de surface des mots. On note que les ensembles de traits que nous utilisons n'ont pas été optimisés pour le français, cette tâche sortant du cadre de notre étude. Les résultats que nous présentons ici ne correspondent donc pas à la performance maximale des systèmes. De plus, nous souhaitons préciser que nous n'entendons pas comparer ici les deux systèmes. Il faudrait utiliser les mêmes ensembles de traits pour discuter a minima de leur implémentation de l'algorithme *MaxEnt*. Les résultats sont donc donnés à titre informatif principalement parce qu'ils sont tous deux utilisés dans la communauté.

Les deux systèmes sont entraînées et évaluées à partir des différents niveaux de correction du FTB. L'ensemble de données FTB+corr. 1 correspond à la correction des étiquettes manquantes par la fréquence (Section 3.1), FTB+corr. 2 correspond à la correction des erreurs d'annotation par la méthode des variations de *n*-grammes (Section 3.2). FTB+corr. 1&2 et FTB+corr. 2&1 correspondent à l'utilisation successive des deux méthodes de correction : corr. 1 puis +corr. 2 et inversement.

Dans la littérature, les méthodes d'étiquetage morpho-syntaxique pour le français ont presque toujours été évaluées à partir d'un découpage du FTB en trois sous-ensembles : 80% pour l'entraînement, 10% pour le développement et 10% pour le test, e.g. (Denis et Sagot, 2010; Constant *et al.*, 2011). Intuitivement, une évaluation fondée uniquement sur 10% des données ne peut pas être représentative du niveau de performance réel d'une méthode. Une première série d'expériences nous a conforté dans cette idée puisque nous avons observé une variation de plus de 2% (en absolu) de la précision en fonction du découpage effectué. La construction incrémentale du FTB ainsi que la nature des documents annotés joue un rôle prépondérant dans ce phénomène. Pour palier ce problème, les résultats que nous présentons dans cette étude ont

3. `http://incubator.apache.org/opennlp/`
4. Nous avons utilisé la macro `naacl2003unknowns` décrite dans (Toutanova *et al.*, 2003).

tous été obtenus en validation croisée en 10 strates, ils ne sont donc pas directement comparables à ceux présentés dans les travaux précédents.

Trois mesures d'évaluation sont considérées comme pertinentes pour nos expériences : la précision sur les tokens, la précision sur les phrases (nombre de phrases dans lesquelles tous les tokens ont été correctement étiquetés par rapport au nombre de phrases total) et la précision sur les mots inconnus. L'écart type (σ) des scores sur les 10 strates est également calculé.

Les résultats sont présentés dans les tables 2 et 3. Les corrections apportées au corpus permettent d'améliorer les scores de précision des méthodes d'étiquetage de manière significative. Ainsi, la précision sur les tokens passe de 96,39 à 97,53 pour le *Stanford POS tagger* et de 95,70 à 97,05 pour *Apache OpenNLP*. De plus, on peut observer que l'écart type des scores calculé sur les 10 strates diminue fortement. Cette mesure est un indicateur de l'amélioration de la stabilité du niveau de performance des systèmes. Une amélioration encore plus importante est observée sur la précision au niveau des phrases, elle passe de 53,05% à 57,05% pour le *Stanford POS tagger* et de 47,56% à 51,67% pour *Apache OpenNLP*. Cette augmentation s'explique par la réduction du nombre de phrases contenant au moins un token auquel aucune étiquette n'a pu être affectée. Concernant la précision au niveau des mots inconnus, la stabilité des scores est normale puisque nous n'introduisons pas de nouveaux tokens dans les données.

Correction	**Stanford POS Tagger**		
	Prec. tokens	Prec. phrases	Prec. inconnus
FTB non corrigé	$96,39\ (\sigma = 0,96)$	$53,05\ (\sigma = 3,71)$	$83,36\ (\sigma = 3,43)$
FTB + corr. 1	$97,52^{\dagger}(\sigma = 0,26)$	$56,76^{\dagger}(\sigma = 2,15)$	$83,52^{\dagger}(\sigma = 3,40)$
FTB + corr. 2	$96,51^{\dagger}(\sigma = 0,89)$	$53,57^{\dagger}(\sigma = 3,61)$	$83,37\ (\sigma = 3,42)$
FTB + corr. 1&2	$97,53^{\dagger}(\sigma = 0,26)$	$57,05^{\dagger}(\sigma = 1,15)$	$83,50\ (\sigma = 3,38)$
FTB + corr. 2&1	$97,53^{\dagger}(\sigma = 0,27)$	$57,02^{\dagger}(\sigma = 2,25)$	$83,51\ (\sigma = 3,40)$

TABLE 2 – Scores de précision obtenus avec le Stanford POS tagger en fonction du niveau de correction du FTB. σ correspond à l'écart type des scores calculé sur les 10 strates. Les scores indiqués par les caractères † sont statistiquement significatifs par rapport au FTB non corrigé ($\rho < 0,01$ avec un t-test de Student).

Correction	**Apache OpenNLP**		
	Prec. tokens	Prec. phrases	Prec. inconnus
FTB non corrigé	$95,82\ (\sigma = 0,95)$	$47,56\ (\sigma = 3,25)$	$85,50\ (\sigma = 1,57)$
FTB + corr. 1	$97,03^{\dagger}(\sigma = 0,26)$	$51,40^{\dagger}(\sigma = 2,04)$	$85,68\ (\sigma = 1,57)$
FTB + corr. 2	$95,94^{\dagger}(\sigma = 0,88)$	$48,08^{\dagger}(\sigma = 3,21)$	$85,50\ (\sigma = 1,60)$
FTB + corr. 1&2	$97,05^{\dagger}(\sigma = 0,26)$	$51,67^{\dagger}(\sigma = 2,13)$	$85,70\ (\sigma = 1,55)$
FTB + corr. 2&1	$97,04^{\dagger}(\sigma = 0,26)$	$51,67^{\dagger}(\sigma = 2,11)$	$85,68\ (\sigma = 1,57)$

TABLE 3 – Scores de précision obtenus avec Apache OpenNLP en fonction du niveau de correction du FTB. σ correspond à l'écart type des scores calculé sur les 10 strates. Les scores indiqués par les caractères † sont statistiquement significatifs par rapport au FTB non corrigé ($\rho < 0.01$ avec un t-test de Student).

La méthode de correction par détection de variations de *n*-grammes permet de ramener davantage d'erreurs candidates lorsque l'on traite des *n*-grammes de taille inférieure à 5. De plus nous avons constaté que pour une taille strictement supérieure à 1, les résultats des étiqueteurs morpho-syntaxiques étaient améliorés. Nous n'avons pas gardé ces résultats car un premier retour au corpus nous conduisait à nous interroger sur la qualité des corrections opérées et par conséquent sur l'amélioration qui pourrait bien être due à un phénomène de lissage des annotations du corpus. Ce dernier point nécessitera une évaluation plus approfondie dans le futur.

5 Travaux connexes

Les méthodes d'étiquetage morpho-syntaxique actuelles, basées sur des modèles probabilistes, offrent un niveau de performance élevé. L'analyse des erreurs restantes suggèrent néanmoins que le gain de précision potentiel venant de meilleurs traits ou d'une méthode d'apprentissage plus performante reste très limité. Les problèmes relevés montrent que les inconsistances et les erreurs d'annotations présentes dans les données d'entraînement et de test sont en partie responsables du palier auquel les méthodes sont confrontées. Partant de ce constat, (Manning, 2011) propose un ensemble de règles manuelles visant à corriger les erreurs d'annotation présentes dans le *Penn Treebank*. Une évaluation comparative de la précision d'un système d'étiquetage morpho-syntaxique sur les données ainsi corrigées a permis de montrer l'efficacité des règles de correction proposées.

Concernant la problématique de détection et de correction automatique d'erreurs d'annotation, la majorité des travaux s'est penchée sur le premier problème avec une attention particulière sur l'étiquetage grammatical (Loftsson, 2009). Outre la question d'annotation d'unité lexicale, le projet DECCA [5] aborde aussi les problèmes de détection d'erreurs d'annotations [6] continues (concernant une séquence de mots), discontinues et de type dépendance.

Quelles que soient les approches et le type d'annotations observé, le principe de détection d'une erreur repose sur la recherche d'annotations inconsistantes au sein du corpus ; c'est-à-dire d'étiquetages différents pour des occurrences comparables du phénomène observé.

Concernant la détection d'erreur d'étiquetage grammatical, cinq approches ont été proposées. (Loftsson, 2009) compare trois méthodes de détection : la première fondée sur la détection de variation de *n*-gramme, la seconde fondée sur l'utilisation de plusieurs étiqueteurs automatiques et la troisième fondée sur de l'analyse syntaxique en constituants. La seconde méthode consiste à utiliser plusieurs étiqueteurs (l'auteur en a utilisé cinq) et à les combiner en utilisant un simple mécanisme de vote (chaque étiqueteur vote pour une étiquette et l'étiquette avec le plus grand nombre de votes est sélectionné). La troisième méthode consiste à exploiter l'étiquette syntaxique produite par l'analyse en constituants pour corriger certaines erreurs éventuelles d'étiquetage grammatical interne. Cette méthode requiert d'identifier dans un premier temps les types d'erreurs d'étiquetage grammatical possibles sous chaque constituant puis d'écrire les règles de correction correspondante. Les résultats de (Loftsson, 2009) montrent que ces méthodes permettent toutes de détecter effectivement des erreurs et qu'elles agissent en complémentarité.

(Loftsson, 2009) rapporte aussi les travaux de (Nakagawa et Matsumoto, 2002) et de (Kveton

5. `http://decca.osu.edu`
6. La nature syntaxique ou sémantique de l'annotation est discutée mais le problème est secondaire.

et Oliva, 2002). (Nakagawa et Matsumoto, 2002) ont utilisé le poids de confiance que leur algorithme de classification (machines à vecteurs de support) utilise pour décider de la classe d'un mot afin de déterminer si la classe assignée était une erreur candidate. Cette méthode est intéressante même si l'entraînement des modèles peut être coûteuse pour de larges jeux d'étiquettes.

(Kveton et Oliva, 2002) décrivent, quant à eux, une méthode semi-automatique pour détecter des *n*-grammes d'étiquettes grammaticales invalides en partant d'un ensemble construit à la main de paires d'étiquettes adjacentes invalides (e.g. un déterminant suivi d'un verbe). La méthode consiste pour chaque bigramme invalide à construire par collecte successive dans les phrases du corpus l'ensemble d'étiquettes pouvant apparaître entre deux étiquettes du bigramme invalide. Tout *n*-gramme d'étiquettes débutant et finissant par les étiquettes d'un bigramme invalide et ayant une étiquette n'appartenant pas à l'ensemble d'étiquettes avérées est considéré comme une erreur potentielle dans un nouveau corpus. Cette méthode requiert d'une part une construction manuelle (par un linguiste) de bigrammes invalides et d'autre part ne permet pas de détecter des *n*-grammes valides d'étiquettes utilisés incorrectement dans certains contextes.

6 Conclusion et perspectives

Nous avons présenté une étude menée sur la détection et la correction automatique des erreurs d'annotation morpho-syntaxique du *French TreeBank*. Deux méthodes ont été utilisées. La première consiste à identifier les mots sans étiquette et leur attribuer celle d'une forme correspondante observée dans le corpus. La seconde méthode utilise les variations de *n*-gramme pour détecter et corriger les anomalies d'annotation. Les résultats que nous avons obtenus montrent que les corrections apportées au corpus permettent d'améliorer de manière significative les scores de précision de deux différentes méthodes d'étiquetage morpho-syntaxique.

Les perspectives de cette étude sont nombreuses. Dans un premier temps, nous souhaitons poursuivre nos travaux en utilisant un jeu d'étiquettes plus étendu. Nous envisageons également d'améliorer la détection des erreurs en utilisant conjointement les variations de *n*-gramme et la combinaison des sorties de plusieurs étiqueteurs (Loftsson, 2009). A plus long terme, nous voulons étudier la possibilité d'utiliser la correction automatique des erreurs d'annotation soit comme une étape préliminaire à la vérification manuelle des annotations, soit comme une alternative. Pour ce dernier point, l'objectif serait d'étendre le FTB par l'ajout de données annotées et corrigées automatiquement.

Les modèles construits à partir des données corrigées pour les étiqueteurs *Stanford POS tagger* et *Apache OpenNLP* sont disponibles à l'adresse : `http://www.lina.univ-nantes.fr/?-TALN-.html`

Références

ABEILLÉ, A., CLÉMENT, L. et TOUSSENEL, F. (2003). Building a treebank for French. *Treebanks : building and using parsed corpora*, pages 165–188.

ARUN, A. et KELLER, F. (2005). Lexicalization in crosslinguistic probabilistic parsing : The case of French. *In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 306–313, Ann Arbor, Michigan. Association for Computational Linguistics.

CONSTANT, M., TELLIER, I., DUCHIER, D., DUPONT, Y., SIGOGNE, A. et BILLOT, S. (2011). Intégrer des connaissances linguistiques dans un CRF : application à l'apprentissage d'un segmenteur-étiqueteur du français. *In Actes de Traitement automatique des langues naturelles (2011)*.

CRABBÉ, B. et CANDITO, M. (2008). Expériences d'analyse syntaxique statistique du français. *In Actes de Traitement automatique des langues naturelles (2008)*.

DENIS, P. et SAGOT, B. (2010). Exploitation d'une ressource lexicale pour la construction d'un étiqueteur morphosyntaxique état-de-l'art du français. *In Actes de Traitement automatique des langues naturelles (2010)*.

DICKINSON, M. et MEURERS, W. D. (2003). Detecting errors in part-of-speech annotation. *In Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL-03)*, pages 107–114, Budapest, Hungary.

GREEN, S., de MARNEFFE, M.-C., BAUER, J. et MANNING, C. D. (2011). Multiword expression identification with tree substitution grammars : A parsing tour de force with french. *In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 725–735, Edinburgh, Scotland, UK. Association for Computational Linguistics.

KVETON, P. et OLIVA, K. (2002). (semi-)automatic detection of errors in pos-tagged corpora. *In COLING*.

LOFTSSON, H. (2009). Correcting a POS-tagged corpus using three complementary methods. *In Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 523–531, Athens, Greece. Association for Computational Linguistics.

MANNING, C. (2011). Part-of-speech tagging from 97linguistics ? *In* GELBUKH, A., éditeur : *Computational Linguistics and Intelligent Text Processing*, volume 6608 de *Lecture Notes in Computer Science*, pages 171–189. Springer Berlin / Heidelberg.

MARCUS, M., MARCINKIEWICZ, M. et SANTORINI, B. (1993). Building a large annotated corpus of English : The Penn Treebank. *Computational linguistics*, 19(2):313–330.

NAKAGAWA, T. et MATSUMOTO, Y. (2002). Detecting errors in corpora using support vector machines. *In COLING*.

TOUTANOVA, K., KLEIN, D., MANNING, C. et SINGER, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. *In Proceedings of the 3rd Conference of the North American Chapter of the ACL (NAACL 2003)*, pages 173–180. Association for Computational Linguistics.