Adaptation de parsers statistiques lexicalisés pour le français : Une évaluation complète sur corpus arborés

Djamé Seddah◇, Marie Candito* et Benoît Crabbé*
◇ Université Paris-Sorbonne * Université Paris 7
LALIC et INRIA (Alpage) UFRL et INRIA (Alpage)

Résumé. Cet article[1] présente les résultats d'une évaluation exhaustive des principaux analyseurs syntaxiques probabilistes dit "lexicalisés" initialement conçus pour l'anglais, adaptés pour le français et évalués sur le CORPUS ARBORÉ DU FRANÇAIS (Abeillé *et al.*, 2003) et le MODIFIED FRENCH TREEBANK (Schluter & van Genabith, 2007).
Confirmant les résultats de (Crabbé & Candito, 2008), nous montrons que les modèles lexicalisés, à travers les modèles de Charniak (Charniak, 2000), ceux de Collins (Collins, 1999) et le modèle des TIG Stochastiques (Chiang, 2000), présentent des performances moindres face à un analyseur PCFG à Annotation Latente (Petrov *et al.*, 2006). De plus, nous montrons que le choix d'un jeu d'annotations issus de tel ou tel treebank oriente fortement les résultats d'évaluations tant en constituance qu'en dépendance non typée. Comparés à (Schluter & van Genabith, 2008; Arun & Keller, 2005), tous nos résultats sont *state-of-the-art* et infirment l'hypothèse d'une difficulté particulière qu'aurait le français en terme d'analyse syntaxique probabiliste et de sources de données.

Abstract. This paper presents complete investigation results on the statistical parsing of French by bringing a complete evaluation on French data of the main based probabilistic lexicalized (Charniak, Collins, Chiang) and unlexicalized (Berkeley) parsers designed first on the Penn Treebank. We adapted the parsers on the two existing treebanks of French (Abeillé *et al.*, 2003; Schluter & van Genabith, 2007). To our knowledge, all the results reported here are state-of-the-art for the constituent parsing of French on every available treebank and invalidate the hypothesis of French being particularly difficult to parse. Regarding the algorithms, the comparisons show that lexicalized parsing models are outperformed by the unlexicalized Berkeley parser. Regarding the treebanks, we observe that a tag set with specific features has direct influences over evaluation results depending on the parsing model.

Mots-clés : Analyse syntaxique probabiliste, corpus arborés, évaluation, analyse du français.

Keywords: Probabilistic parsing, treebanks, evaluation, French parsing.

[1]Ce travail est soutenu par l'ANR programme DEFI 2008 dans le cadre du projet SEQUOIA.
Nous tenons à remercier Abishek Arun, Josef Van Genabith, Alexis Nasr et Natalie Schluter pour nous avoir permis d'utiliser leurs ressources ainsi que Marwan El Markour et Rosa Stern pour leur aide.

1 Introduction

Depuis la mise à disposition progressive du Corpus Arboré de Paris 7 (FTB, (Abeillé *et al.*, 2003)), différents travaux précurseurs portant tant sur l'extraction de grammaires d'arbres adjoints (Dybro-Johansen, 2004) que sur l'entraînement d'analyseurs statistiques (Arun & Keller, 2005) sont apparus et démontrent la faisabilité de telles tâches sur le FTB. Néanmoins, l'absence d'annotations fonctionnelles permettant de pallier l'aspect plat des structures syntaxiques du FTB et l'état préliminaire du corpus disponible à cette époque compliquent ces tâches et rendent délicate toute comparaison avec les travaux récents de (Crabbé & Candito, 2008) et ceux ayant motivé la création du "Modified French Treebank" (Schluter & van Genabith, 2007; Schluter & van Genabith, 2008).

Dans cet article, nous présentons les résultats d'une évaluation extensive de cinq modèles de parsing probabiliste sur les deux corpus arborés du français, qui montrent que (1) des résultats *state of the art* sont atteints sur toutes les versions du corpus, par conséquent le problème de la source de données pour l'analyse syntaxique du français est résolue et (2) que le débat porté par N.Schluter et A.Arun, sur l'utilité d'algorithmes lexicalisés pour le français, est loin d'être tranché.

Dans un premier temps, section 2, nous présentons nos corpus de travail, puis, section 3, nous introduisons brièvement les modèles d'analyses avant de présenter notre protocole expérimental section 4. Enfin, nous discutons les résultats présentés section 5 puis nous concluons.

2 Treebanks pour le français

Cette section présente un bref aperçu des corpus pour lesquels nous présentons des résultats : le Corpus Arboré du Français (FTB, (Abeillé *et al.*, 2003)) et le *Modified French Treebank* (MFT, (Schluter & van Genabith, 2007)).

Le Corpus arboré du français (FTB)

Le FTB est le premier corpus arboré (ie. treebank) annoté et corrigé manuellement disponible pour le français. Les annotations sont morphologiques et syntaxiques, les secondes incluant des annotations fonctionnelles pour les dépendants verbaux (cf. (Abeillé *et al.*, 2003) pour plus de détails). La version utilisée pour ce travail contient 12351 phrases, 385 458 occurrences de tokens et présente une longueur moyenne de 27 tokens. Comparativement à une longueur moyenne de 24 tokens pour le Penn Treebank (i.e. PTB, (Marcus *et al.*, 1994)) qui contient près de 44 000 phrases pour plus d'un million de tokens, cela rend la tache d'analyse syntaxique plus délicate. Comme d'autres treebanks (par exemple Tiger, Negra pour l'allemand), le FTB propose une structure hiérarchique plus plate que celle du PTB. En effet, basée sur une représentation X-barre, celle-ci permet une distinction configurationnelle des syntagmes sous-catégorisés par le verbe tandis que seule l'annotation fonctionnelle permet cette distinction dans le cas du FTB. Ceci est lié, entre autres, au choix de ne pas inclure de noeuds de type VP dans les annotations syntaxiques du FTB et a en partie conduit les auteurs de (Schluter & van Genabith, 2007) à ré-annoter une partie du corpus à leur disposition dans une optique d'extraction de grammaires LFG.

le "Modified French Treebank" (MFT)

Le MFT (Schluter & van Genabith, 2007; Schluter & van Genabith, 2008) est un sous-ensemble de 4739 phrases tirées du FTB initial, ré-annotées semi-automatiquement et corrigées à la main. Ses auteurs ont introduit deux différences formelles par rapport à la source : des différences structurelles et des modification du jeu d'annotation. D'un point de vue structurel, les trans-

formations principales incluent une stratification accrue des règles sous-jacentes au treebank (comme l'introduction d'un noeud VP) et mettent l'accent sur une modification du schéma de coordination (celui-ci ajoutant désormais à l'étiquette COORD le label du noeud coordonné). Ces modifications permettent une meilleur adéquation à l'architecture décrite dans (Cahill *et al.*, 2004) et permettent de réduire la taille de la grammaire extraite du treebank. Par ailleurs, le MFT inclut aussi un affinage du jeu d'annotation via l'ajout de certaines informations morphologiques sur les labels des nœuds (tels que des traits de mode supplémentaires sur les noeuds VP et VN) et par l'affinement du jeu d'étiquettes fonctionnelles.
Finalement, une phase d'*error mining* et une correction manuelle extensive ont été appliquées sur le corpus.
Le tableau 1 synthétise un certain nombre d'éléments de comparaison entre les deux treebanks sous leur forme canonique. Les nombres reportés sont calculés en fonction des catégories syntaxiques de bases, sans annotations fonctionnelles et sans annotations morpho-syntaxiques sur les parties du discours (eg. pas de traits "genre" ni "nombre"). La comparaison du nombre moyen de branchements par nœuds (2.60 pour le FTB et 2.11 pour le MFT) démontre la stratification plus élevée du MFT.

Carac.	FTB	MFT
Taille (phrases)	12351	4739
Longueur moy. des phrases	27.48	28.38
Nbre moy. de branch/noeuds	2.60	2.11
Taille de la grammaire (hors règles lex.)	14874	6944
Symb. non-terminaux	13	39
POS tags	15	27

TAB. 1 – Statistiques sur Treebanks

3 Algorithmes d'analyse syntaxique probabiliste

Les grammaires hors-contexte probabilistes (PCFG) sont le formalisme de base pour l'analyse syntaxique (ie. parsing) probabiliste et leur extraction est aisée de par la nature fondamentalement hors-contexte des annotations syntaxiques d'un treebank(Jurafsky *et al.*, 2000). Néanmoins, de la faible puissance générative du modèle découlent deux problèmes majeurs pour le parsing à partir de grammaires extraites de treebank :**(a) Les hypothèses d'indépendances du modèle PCFG sont trop fortes**, **(b) Les probabilités lexicales ne sont pas prises en compte par le modèle de base**.
Des techniques se projetant dans différents paradigmes de parsing, ont eté proposées, principalement pour l'anglais, pour résoudre ces deux problèmes. Nous nous proposons d'explorer ces techniques en les appliquant sur le français via deux classes d'analyseurs. Une première classe non lexicalisée qui tente de répondre au problème (a) et différents modèles lexicalisés répondant aux problèmes (a) et (b).

Algorithmes lexicalisés : L'idée sous-jacente aux algorithmes lexicalisés est de modéliser les dépendances lexicales entre un gouverneur et ses dépendants afin d'améliorer les choix d'attachements (Jurafsky *et al.*, 2000). L'idée semble évidente mais bien qu'il ait été maintes fois prouvé que la lexicalisation était utile pour le parsing du PTB (Collins, 1999; Charniak, 2000), la question de son adéquation à d'autres langues s'est posée pour l'allemand (Dubey & Keller, 2003) et pour le français (Arun & Keller, 2005). Pour ce dernier, les auteurs défendent l'idée

que le parsing du français tire bénéfice de la lexicalisation mais que la *platitude* du treebank réduit son impact. Notons que ce point a été remis en cause par (Schluter & van Genabith, 2007). En effet, les auteurs maintiennent qu'un schéma d'annotation amélioré ainsi qu'une plus grande homogénéité dans le treebank participent à l'obtention de résultats *state-of-the-art*.
Ce débat sur le français et la lexicalisation s'est concentré sur l'implémentation des modèles 1 et 2 de Collins par (Bikel, 2002) en tant qu'instance d'algorithmes lexicalisés. Le modèle génératif de Collins ayant été extrêmement optimisé pour les annotations du PTB (Bikel, 2004), les tentatives d'adaptation vers d'autres langues se sont montrées plus ou moins fructueuses (en particulier pour l'allemand où un modèle PCFG s'est avéré plus efficace que le modèle 2 de Collins, même avec un modèle de parsing enrichi (Dubey & Keller, 2003)).
C'est pourquoi afin d'apporter des éléments de réponses supplémentaires, nous avons adapté, outre les modèles de Collins via l'adaptation au FTB de leur implémentation par (Bikel, 2002), le parser de Charniak (Charniak, 2000) ainsi que le parser STIG de David Chiang (Chiang, 2000).

Algorithmes non lexicalisés : Comme instance de ce paradigme, le dernier parser que nous utilisons est le parser de Berkeley (ie. BKY, (Petrov *et al.*, 2006)). Son algorithme est une évolution des principes de transformation de treebank visant à réduire les hypothèses d'indépendance propres aux PCFG (Johnson, 1998; Klein & Manning, 2003). Les transformations de treebank peuvent être de deux types : (1) modification de structures et (2) modification du jeu d'annotation. BKY se concentre sur le second point en considérant la recherche d'un jeu d'annotation des symboles non-terminaux comme un problème d'apprentissage semi-supervisé visant à apprendre une PCFG à Annotations Latentes (PCFG-LA). (Crabbé & Candito, 2008) rapporte les scores les plus élevés d'évaluation en constituants sur le FTB en ayant adapté ce parser pour le français puis construit un jeu d'annotations optimisant ses résultats.

4 Protocole expérimental

4.1 Paramétrage des parsers

Configuration : Dans le cas de BKY, suivant en cela (Crabbé & Candito, 2008), nous l'utilisons avec une markovisation horizontale $h = 5$ et 5 cycles *split/merge*. Toute l'information nécessaire à l'apprentissage de BKY est contenue dans le treebank, aucune heuristique n'est utilisée excepté pour le traitement des mots inconnus qui suit celui de (Arun & Keller, 2005). Tous les autres parsers sont utilisés dans leurs configurations de base et n'ont subi que des modifications liées aux nouveaux jeux d'annotations propres aux treebanks du français. Notons enfin que les jeux d'annotations n'ont pas été modifiés spécifiquement pour tel ou tel parser, ainsi les auxiliaires, contrairement à (Charniak, 2000), ne reçoivent pas de traitements particuliers.
Adaptation Morphologique et typographique : Dans le cas des parsers lexicalisés, nous avons automatiquement converti les parties du discours associées aux marques de ponctuation au format du PTB. Le traitement morphologique pour les mots inconnus est le même pour les modèles de Collins que pour BKY. Les mots inconnus sont *clusterisés*, à l'aide d'indices typographiques et morphologiques, si leur fréquence est inférieure à 6 sauf dans le cas de CHARNIAK où tous les mots sont pris en compte mais subissent un lissage lexical afin d'atténuer la dispersion de données.
Table de percolation des têtes et distinction argument-adjoint : Tous les parsers lexicalisés que nous utilisons font usage d'une table de percolation de tête (ie. *headrules*) qui utilise des informations configurationnelles pour déterminer dynamiquement la tête d'un noeud donné en

fonction des catégories de ses fils (Collins, 1999). Par conséquent, l'adaptation de ces parsers au français nécessite de construire une telle table. Nous avons ainsi adapté celles construites par (Dybro-Johansen, 2004) à des fins d'extractions de grammaires d'arbres adjoints à partir du FTB originel. Comme le modèle 2 de Collins et le modèle STIG nécessitent de pouvoir distinguer entre arguments et adjoints (pour apprendre les probabilités des cadres de sous-catégorisations dans le cas de Collins et pour extraire les arbres initiaux de la grammaire TIG dans celui de STIG), nous avons implémenté une table de distinction argument-adjoint (désormais TDAA) basée sur les annotations fonctionnelles. C'est l'une des principales différences entre nos expérimentations et celles de (Arun & Keller, 2005; Dybro-Johansen, 2004) où les auteurs, n'ayant pas de corpus avec annotations fonctionnelles, ont dû construire des TDAA uniquement basées sur les catégories syntaxiques d'un corpus extrêmement plat.

Détails d'implémentation Notons que dans le cas du parser STIG, le fait de n'avoir pas accès à une TDAA le conduit à extraire une grammaire où presque tous les arbres ont une structure "filaire" composée uniquement du chemin entre un élément lexical et sa projection maximale (que nous appelons *spine*[2]). Cette configuration particulière permet au modèle probabiliste STIG, se découpant entre les probabilités associées aux schémas d'arbres élémentaires et celles des ancres, de produire une grammaire moins éparpillée que le modèle STIG standard. Précisons enfin que le transcodage des tables vers les formats attendus par les parsers est entièrement automatisé.

En guise d'émulation "brutale" du modèle 1 de Collins, nous utilisons le paramétrage standard du modèle 2 fourni par l'implémenation sans informations de sous-catégorisation. Par ailleurs, en utilisant un jeu de paramètres non standard visant à modifier la façon dont le modèle génératif tient compte des non-terminaux modifieurs, nous obtenons des résultats significativement meilleurs que ceux du modèle 2 sur le français. Dans un cas (le MODÈLE X) tous les modifieurs précédant la tête sont inclus dans le modèle alors que seul le modifieur, préalablement clusterisé, qui la précède est inclus dans le MODÈLE 2.

4.2 Protocole d'évaluation

Pour chaque expérience, nous donnons les résultats selon un découpage classique des treebanks en 3 parties (entraînement, développement et test) de respectivement 80, 10 et 10% de la taille totale du corpus. Les corpus de test et de développement sont respectivement les deux premières tranches de 1235 phrases du FTB et sont prédéfinis pour le MFT. Dans tous les cas, les mots composés sont fusionnés dans une phase de préprocessus. Les parsers reçoivent en entrée du texte nu, excepté le parser STIG qui n'accepte que des entrées étiquetées et pour lequel nous avons entraîné le tagger TNT (Brants, 2000) sur les treebanks.[3]

Métriques d'évaluation : Nous utilisons la métrique standard PARSEVAL [4] ainsi qu'une évaluation en dépendance non typée, décrite comme une métrique plus neutre que PARSEVAL face au jeu d'annotation (cf.(Rehbein & van Genabith, 2007)). L'évaluation des dépendances non typées est faite selon l'algorithme de (Lin, 1995) et utilise les *headrules* de Dybro-Johansen. Le F-score en dépendances non-typées donne le pourcentage des tokens hors ponctuation qui

[2] A ne pas confondre avec le *spine* dans le modèle TAG qui est le chemin entre un noeud pied et la racine d'un arbre auxiliaire (Joshi, 1987).

[3] Les performances de cet étiqueteur sont du même ordre que celles de l'étiqueteur interne des autres parsers, avec une précision d'étiquetage allant de 97.33% sur le FTB avec tagset minimal pour BIKEL (modèle 1) et 97.21% pour STIG (spinal) avec entrées TNT.

[4] Implémentée par le programme classique EVALB avec les paramètres standard de Collins et calculée sur les phrases de longueur < à 40 mots.

reçoivent la tête correcte.

Baseline : Comparaison sur les jeux d'annotations minimum Nous avons comparé tous les parsers sur 2 différentes instances du FTB et du MFT et afin d'établir une *baseline*, les jeux d'annotations (ie. tagset) des treebanks sont convertis vers un tagset minimal ne contenant que les catégories syntaxiques de base sans aucune autre information que les étiquettes fonctionnelles, utilisées uniquement dans le cas précis des parsers STIG-pure et des modèles de Collins. Notons qu'ici nous ne cherchons pas à comparer la forme des treebanks ou leurs *parsabilités* intrinsèques, il s'agit simplement d'établir un tour d'horizon des parsers sur des treebanks dénués de toute optimisation de leurs tagsets. Dans tous les cas, nous observons que BKY présente des performances supérieures aux autres dans toutes les métriques (cf.Tableau 2), ce qui confirme les résultats observés dans (Crabbé & Candito, 2008). le parser STIG, dans ses deux modes de fonctionnement *pur* et *spinal*, ne présente pas de différence statistiquement signifiante en métrique PARSEVAL[5] au moins dans les résultats PARSEVAL. C'est pourquoi dans un souci d'espace nous ne présentons par la suite que les résultat en mode STIG-spinal.

		FTB-min	MFT-min
COLLINS MX	PARSEVAL	81.65	79.19
	UNLAB. DEP	88.48	84.96
COLLINS M2	PARSEVAL	80.1	78.38
	UNLAB. DEP	87.45	84.57
COLLINS M1	PARSEVAL	77.98	76.09
	UNLAB. DEP	85.67	82.83
CHARNIAK	PARSEVAL	82,44	81.34
	UNLAB. DEP	88.42	84.90
CHIANG-SPINAL	PARSEVAL	80.66	80.74
	UNLAB. DEP	87.92	85,14
BKY	PARSEVAL	84,93	83.16
	UNLAB. DEP	90.06	87.29
CHIANG-PUR	PARSEVAL	80.52	79.56
	UNLAB. DEP	87,95	85.02

TAB. 2 – F_1 scores des parsers lexicalisés et non lexicalisés sur Treebank avec tagset minimal

5 Evaluation des analyseurs en fonction des variations des jeux d'annotations

Dans (Crabbé & Candito, 2008) les auteurs ont présenté des expériences visant à déterminer un jeu d'annotation (i.e. tagset), nommé FTB-CC ici et TREEBANK+ dans l'article, maximisant les performances du parser de Berkeley sur le FTB avec un F_1-score(<=40) de 86.41%. Ce tagset inclut des informations de mode pour les verbes (ie. indicatif, impératif, participe passé, etc.) et certains traits de sous-catégorie (cf. (Crabbé & Candito, 2008), Table 2). L'impact de diverses variations de tagsets appliquées au FTB, qui n'a pas été conçu dans une optique de *parsing*, a ainsi été testé via des mesures de constituance comme indicateur de performance.
Sachant que le MFT a par contre été conçu dans une optique de maximisation des performances d'un analyseur de grammaires LFG, donc visant à produire des dépendances syntaxiques profondes, induites à partir de sortie d'analyseurs statistiques (Schluter & van Genabith, 2008), il offre des performances néanmoins étonnantes au vu de sa taille réduite. L'influence de son tagset et de ses modifications structurelles sont déterminantes et il aurait été intéressant de vérifier

[5]avec une p-value en F-score de 0.32

leur impact sur davantage de données.
Malheureusement, des modifications semi-automatiques du MFT (en particulier celles apportées au schéma de coordination) ne peuvent pas être reproduites de façon réversible et automatique. Toutefois, si nous ne pouvons pas réellement évaluer l'influence de la structure, nous pouvons évaluer celle d'un tagset particulier appliqué à un autre treebank à l'aide d'outils de conversions. A cette fin, les tagsets maximisant les résultats PARSEVAL sont extraits de leurs treebanks respectifs (le tagset CC pour le FTB et le tagset SCHLU pour le MFT) et appliqués sur l'autre treebank. Nous avons donc deux tagsets pour chaque treebank sur lesquels nous évaluons chaque parser en dépendance non-typée et en constituance. La table 3 présente les résultats en surlignant les meilleurs scores pour chaque paire de treebanks.

Parser	**Parseval**		**Dependency**		**Parseval**		**Dependency**	
	MFTCC	MFTSCH.	MFTCC	MFTSCH.	FTBCC	FTBSCH.	FTBCC	FTBSCH.
Collins (MX)	80.2	80.96	85.97	**87.98**	82.52	82.65	88.96	89.12
Collins (M2)	78.56	79.91	84.84	87.43	80.8	79.56	87.94	87.87
Collins (M1)	74	78.49	81.31	85.94	79.16	78.51	86.66	86.93
Charniak	82.5	82.66	86.45	86.94	84.27	83.27	89.7	89.67
Chiang (Sp)	82.6	81.97	86.7	87.16	81.73	81.54	88.85	89.02
Bky	**83.96**	**82.86**	**87.41**	86.87	**86.02**	**84.95**	**90.48**	**90.73**

TAB. 3 – Résultats d'évaluation MFT-CC vs MFT-SCHLU et FTB-CC vs FTB-SCHLU

Les résultats de ces expériences, Table 3, confirment la tendance visible dans le cas d'un parsing avec tagset minimal (cf. Table 2), à savoir que BKY présente toujours les scores les plus élevés quelque soit la métrique. Notons que les évaluations en dépendances non-typées sont systématiquement meilleures sur le tagset SCHLU que sur le tagset CC. On peut l'expliquer par une plus grande précision des *headrules* sur ce tagset. En effet, ces règles ayant été générées à partir de méta-descriptions[6] , leurs couvertures et leurs précisions globales sont plus élevées. On a, par exemple, 18 règles pour FTB-CC et 43 pour FTB-SCHLU.
Comme prévu au regard des scores sur le PTB, le classement PARSEVAL des parsers lexicalisés donne à CHARNIAK les meilleurs performances quel que soit le tagset, en ne considérant pas le score de STIG-spinal sur le MFT-CC qui témoigne d'une variation non statistiquement significative avec le score de CHARNIAK sur ce treebank.[7] En revanche, l'évaluation en dépendance des parsers lexicalisés est différentes selon le treebank. Dans le cas du FTB, CHARNIAK présente les meilleurs scores, tandis que le modèle STIG-spinal a de meilleures performances sur les MFT-SCHLU et MFT-CC. Notons que les variations du modèle 2 de Collins présentent des résultats élevés en dépendance sur le MFT-SCHLU alors que leurs scores parseval sont les plus faibles. Les faibles scores des modèles de Collins de base en constituance peuvent s'expliquer par la dispersion de données accrue apportée par les annotations fonctionnelles, en particulier sur des corpus de taille réduite.

6 Discussion

Comme nous l'avons déja dit dans l'introduction, les travaux précurseurs sur le FTB ont été initiés par (Dybro-Johansen, 2004) dans une optique d'extraction de grammaire TAG ; bien qu'elle n'y reporte pas de résultats d'analyse syntaxique, les mêmes problèmes de distinction

[6]Un label COORD se réécrit par exemple COORD_vfinite, COORD_sint, etc.
[7]P-value élevée de 0.1272 en précision et 0.06 en rappel.

entre compléments et adjoints que (Arun & Keller, 2005) se sont posés. C'est l'aspect très plat du treebank ainsi que l'absence d'annotations fonctionnelles dans cette version distribuée en 2004 qui ont conduit Arun à modifier le jeu d'annotation (par exemple VNG pour distinguer les noeuds VN qui dominent un verbe sous-catégorisant des clitiques) et à enrichir le modèle génératif de Collins afin d'améliorer les performances globales de l'analyseur.
La question se pose de savoir si ces modifications se justifiaient sur le treebank initial étant donné que les résultats de nos analyseurs entraînés sur le corpus initial sont supérieurs non seulement à ceux reportés dans (Arun & Keller, 2005)[8] mais aussi à ceux obtenus en utilisant leur propre implémentation du modèle 2 de Collins entraîné avec notre table de percolation de tête et avec leur propre TDAA.

PARSER	FTBARUN	MFTSCHLU
Arun (acl05)	**80.45**	-
Arun (emnlp)	**81.08**	-
Schluter	-	**79.95**
Collins (Mx)	81.5	80,96
Collins (M2)	79.36	79,91
Collins (M1)	77.82	-
Charniak	82.35	82,66
Chiang (Sp)	80.94	81,86
Bky	84.03	82.86

TAB. 4 – Scores Parseval sur le FTB utilisé par Arun et sur le MFT

Nous sommes aussi directement comparables avec (Schluter & van Genabith, 2007) dont le meilleur F_1 score PARSEVAL en texte nu est de 79.95 quand le nôtre est de 82.86 sur le MFT (cf. Table. 4).

Concernant les faibles résultats du modèle 2 de Collins dans presque tous les cas de figure, cela est selon nous dû à l'éparpillement provoqué par l'ajout d'annotations fonctionnelles dans des petits treebanks. De plus, conçus pour l'anglais, son modèle s'exporte manifestement moins bien dans des treebanks moins hiérarchisés que le PTB. Cette observation rejoint celles émises par (Corazza *et al.*, 2004) dans le cas d'une adaptation de ce modèle à l'italien sur de très petits treebanks (moins de 3000 phrases). En revanche, nous ne partageons pas le point de vue communément admis sur un relatif échec des modèles lexicalisés ; à notre connaissance seuls les modèles de Collins, via leur implémentation de BIKEL ont été adaptés à des langues européennes. Les performances comparées du modèle de Charniak face à celui de BKY s'évaluent selon un ordre de grandeur similaire à celui connu pour le parsing du la section 23 du PTB. Par manque de place, nous ne pouvons inclure qu'un graphique représentant la courbe d'apprentissage des parsers, n'utilisant pas de TDAA, en mode Perfect-tagging sur le FTB-CC (fig. 1), mais celle-ci montre que la courbe d'apprentissage de CHARNIAK est quasiment parallèle à celle de BKY tandis que les modèles de Collins (*ici le modèle X sans TDAA a été utilisé*) et STIG ont des courbes qui se confondent quasiment et qui plafonnent très vite.[9] Ces deux modèles ayant des *back-off* extrêmement similaires, on peut se demander (1) si ce n'est pas eux qu'on compare en réalité et (2) si la petite taille des corpus autres que le PTB ne pousse pas la communauté à

[8] Les résultats actualisés sont disponibles via l'url suivante :
(http ://homepages.inf.ed.ac.uk/s0343799/acl2005slides.pdf).

[9] Nous sommes bien sûr conscients que les valeurs de cette courbe sont aussi fonction du nombre de nouvelles productions amenées par l'accroissement du lexique ; dans ce cas il faudrait aussi comparer les modes de "pruning" de ces modèles.

considérer les modèles lexicalisés, à travers les seuls modèles de Collins, comme inadéquats à d'autres langues que l'anglais du type *Wall Street Journal*.

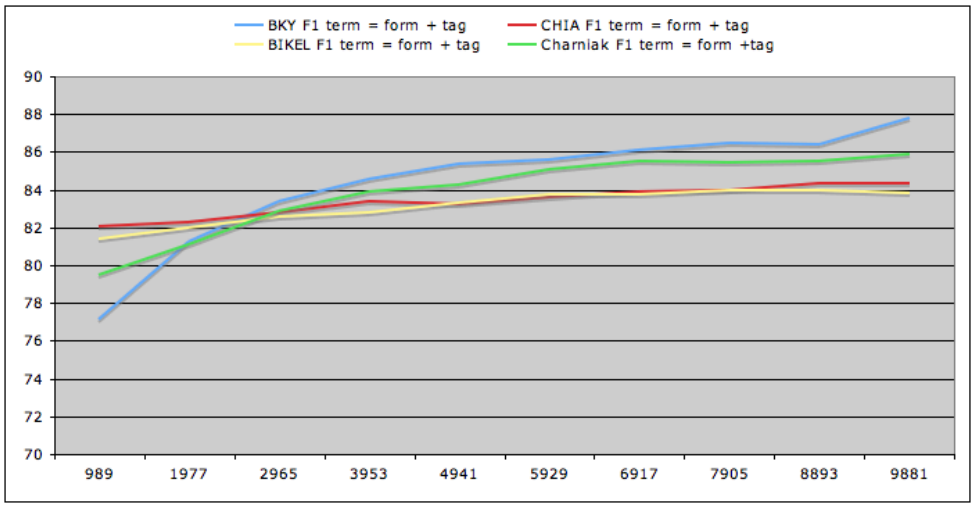

FIG. 1 – Courbe d'apprentissage sur FTB-CC en mode perfect-tagging

7 Conclusion

Nous avons présenté des résultats de parsing statistique lexicalisé et non lexicalisé sur tous les treebanks du français à notre disposition. Ces résultats *state of the art* confirment la maturité du FTB en cette matière et soulignent l'influence du jeu d'annotation sur les performances des analyseurs. Par ailleurs, par le test de multiples analyseurs, nous avons montré que le débat sur les bénéfices de la lexicalisation pouvait bénéficier de l'inclusion de différents modèles lexicalisés.

Références

ABEILLÉ A., CLÉMENT L. & TOUSSENEL F. (2003). *Building a Treebank for French*, In *Treebanks*. Kluwer : Dordrecht.

ARUN A. & 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*, p. 306–313, Ann Arbor, MI.

BIKEL D. M. (2002). Design of a multi-lingual, parallel-processing statistical parsing engine. In *Proceedings of the second international conference on Human Language Technology Research*, p. 178–182 : Morgan Kaufmann Publishers Inc. San Francisco, CA, USA.

BIKEL D. M. (2004). Intricacies of Collins' Parsing Model. *Computational Linguistics*, **30**(4), 479–511.

BRANTS T. (2000). Tnt – a statistical part-of-speech tagger. In *Proceedings of the 6th Applied NLP Conference (ANLP)*, Seattle-WA.

CAHILL A., BURKE M., O'DONOVAN R., VAN GENABITH J. & WAY A. (2004). Long-Distance Dependency Resolution in Automatically Acquired Wide-Coverage PCFG-Based LFG Approximations. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, p. 320–327, Barcelona, Spain.

CHARNIAK E. (2000). A maximum-entropy-inspired parser. In *Proceedings of the 1st Annual Meeting of the North American Chapter of the ACL (NAACL)*, Seattle.

CHIANG D. (2000). Statistical parsing with an automatically-extracted tree adjoining grammar. *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, p. 456–463.

COLLINS M. (1999). *Head Driven Statistical Models for Natural Language Parsing*. PhD thesis, University of Pennsylvania, Philadelphia.

CORAZZA A., LAVELLI A., SATTA G. & ZANOLI R. (2004). Analyzing an Italian treebank with state-of-the-art statistical parsers. In *Proc. of the Third Third Workshop on Treebanks (TLT 2004) and Linguistic Theories*.

CRABBÉ B. & CANDITO M. (2008). Expériences d'analyse syntaxique statistique du français. In *Actes de la 15ème Conférence sur le Traitement Automatique des Langues Naturelles (TALN'08)*, p. 45–54, Avignon.

DUBEY A. & KELLER F. (2003). Probabilistic parsing for german using sister-head dependencies. In *In Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, p. 96–103.

DYBRO-JOHANSEN A. (2004). Extraction automatique de grammaires à partir d'un corpus français. Master's thesis, Université Paris 7.

JOHNSON M. (1998). PCFG models of linguistic tree representations. *Computational Linguistics*, **24**(4), 613–632.

JOSHI A. K. (1987). Introduction to tree adjoining grammar. In A. MANASTER-RAMER, Ed., *The Mathematics of Language* : J. Benjamins.

JURAFSKY D., MARTIN J., KEHLER A., VANDER LINDEN K. & WARD N. (2000). *Speech and language processing : An introduction to natural language processing, computational linguistics, and speech recognition*. MIT Press.

KLEIN D. & MANNING C. D. (2003). Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, p. 423–430 : Association for Computational Linguistics Morristown, NJ, USA.

LIN D. (1995). A dependency-based method for evaluating broad-coverage parsers. In *International Joint Conference on Artificial Intelligence*, p. 1420–1425, Montreal.

MARCUS M. P., SANTORINI B. & MARCINKIEWICZ M. A. (1994). Building a large annotated corpus of english : The penn treebank. *Computational Linguistics*, **19**(2), 313–330.

PETROV S., BARRETT L., THIBAUX R. & KLEIN D. (2006). Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, Australia : Association for Computational Linguistics.

REHBEIN I. & VAN GENABITH J. (2007). Treebank Annotation Schemes and Parser Evaluation for German. In *Proceedings of the 16th Nordic Conference of Computational Linguistics NODALIDA-2007*, Tartu, Estonia.

SCHLUTER N. & VAN GENABITH J. (2007). Preparing, restructuring, and augmenting a french treebank : Lexicalised parsers or coherent treebanks ? In *Proceedings of PACLING 07*.

SCHLUTER N. & VAN GENABITH J. (2008). Treebank-based acquisition of lfg parsing resources for french. In E. L. R. A. (ELRA), Ed., *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco.