Introduction aux étiqueteurs grammaticaux (taggeurs)

D'après (Manning and Schütze 1999), chap. 10

Philippe Langlais

felipe@iro.umontreal.ca

January 28, 2016

Plan

Qu’est-ce qu’un taggeur ?

Approches de base

- approche markovienne
- cas particulier: chunking
- approche transformationnelle

But d'un taggeur

But: associer chaque mot d'une phrase à une **étiquette grammaticale** (ou tag) comme: ADJ, NOMC, NOMP, DET, etc.

- on parle également d'étiquettes **Part Of Speech** (POS).

mot	tag	mot	tag
la	Dete-dart-ddef-femi-sing	à	Prep
séance	NomC-femi-sing	15	Quan-femi-plur-qdef
est	Verb-IndPré-sing-p3	h	NomC-femi-plur
ouverte	Verb-ParPas-femi-sing	43	Quan-masc-plur-qdef

Pourquoi est-ce intéressant ?

- analyse partielle (shallow parsing)
- des taux de bon étiquetage *raisonnables* (supérieurs à 95%),
- utile dans certaines applications comme:
  - l'extraction d'information (information extraction)
  - la réponse automatique à des questions (question answering)

Idée: les POS suffisent souvent à identifier des groupes syntaxiques simples comme les groupes nominaux.

Exemple (fictif) d'extraction d'information

Tâche: remplir des formulaires découverte/auteurs à partir de textes.

> Kuhn Jeff , a physicist at the Institute for Astronomy at the University of Hawaii, and his colleagues may have found evidence of some kind of emission process in the plane of the planets.

champ	information
découvreur	Kuhn Jeff and his colleagues
status	physicist at the Institute for Astronomy at the University of Hawaii
découverte	some kind of emission process in the plane of the planets

Le jeu d'étiquettes (le *tag set*)

- Dépend de l'application et de la précision requise.
- En général un ensemble de 40 à 400 étiquettes.
- Au RALI, un étiqueteur du français été entraîné sur un jeu de 330 étiquettes. En voici quelques unes:

tag	signification	exemple
NomC-masc-sing	Nom commun masculin singulier	haricot
NomC-femi-sing	Nom commum féminin singulier	poire
Verb-IndImp-sing-p3	verbe à l'indicatif imparfait, 3ème personne du singulier	voulait
AdjQ-masc-plur	adjectif qualificatif masculin pluriel	nombreux
ConC	conjonction de coordination	et
ConS	conjonction de subordination	que

Tagset populaire

(Charniak 1993) p.3

POS	signification	exemples
noun	nom commun	dog, equation, concerts
prop	nom propre	Alice, Romulus
pro	pronom	I,you,it,they,them
pos	possessif	my, your
verb	verbe	is, touch, went, remitted
adj	adjectif	red, large, remiss
det	article	the, a, some
prep	préposition	in, to, into
conj	conjonction	and, but, since
aux	auxiliaire	be, have
modal	vb. modaux	will, can, must, should
adv	adverbe	closely, quickly
wh	wh-mouvements	who, what, where
punc	ponctuation	. ? !

Est-ce difficile de tagger?

La belle ferme le voile

- ART NOMC VERB ART NOMC ▷ une jolie femme qui ferme un voile.
- ART ADJQ NOMC PRO VERB ▷ une ferme voile la vue de la chose dont on fait mention par *le*.

belle ▷ *adjectif féminin singulier*
▷ *nom commun féminin singulier*.

ferme ▷ *adjectif singulier* (féminin ou masculin)
▷ *nom commun féminin singulier*
▷ *verbe* (indicatif présent (1,3-ps), impératif présent (2ps), subjonctif présent (1,3-ps)).

voile ▷ *nom commun singulier* (féminin ou masculin)
▷ *verbe* (indicatif présent (1,3-ps), impératif présent (2ps), subjonctif présent (1,3-ps))

Est-ce difficile de tagger ?

- Il existe cependant de nombreux mots qui ne sont étiquetables que par un seul tag:

mot	tag
âge	NomC-masc-sing
âne	NomC-masc-sing
ânerie	NomC-fem-sing
éducatif	AdjQ-masc-sing
électoraux	AdjQ-masc-plur
zyeutera	Verb-IndFutur-sing-p3

- Peut dépasser 50% des types d'un grand corpus

Quelle information utiliser pour tagger ?

Certaines séquences sont plus fréquentes que d'autres

- Il est plus fréquent en français d'avoir la séquence:
  ART ADJ NOMC (le blanc manteau de neige) que
  ART ADJ VERB (est-ce même possible ?)
- Un taggeur qui se baserait sur cette information devrait normalement associer l'étiquette **NOMC** à ébauche plutôt que l'étiquette **VERB** (3ème personne du singulier de l'indicatif présent ou subjonctif) dans la phrase: la belle ébauche.

- L'information du contexte n'est pas forcément fiable

En pratique, cette information seule ne suffit pas (taux de 77%)

Quelle information utiliser pour tagger ?

Le mot lui-même

- belle est probablement plus fréquemment employé en français comme un adjectif que comme un nom commun.

- (Charniak 1993) reporte qu'un tagger simple qui étiquette un mot par son étiquette la plus fréquente (et qui étiquette NOM-PROPRE un mot inconnu) permet d'obtenir des taux d'étiquetage de l'ordre de 90%

- requiert l'étiquette la plus fréquente d'un mot (listée dans certains dictionnaires)

Est-ce qu'un taux de 95% est un bon taux ?

- 5 erreurs tous les 100 mots.
- 1 phrase $\sim$ 20 mots $\Longrightarrow$ une erreur par phrase
  plusieurs erreurs peuvent intervenir dans la même phrase

- bien sûr, tout dépend de l'application...

- Il est toujours difficile de comparer des taggeurs entraînés sur des corpus différents:
  - pourcentage de mots qui possèdent plus d'une étiquette dans le train?
  - taille du vocabulaire?
  - tagset?
  - taux de mots inconnus?

  Voir l'action de recherche GRACE (Adda, Mariani, Paroubek, Rajman, and Lecomte 1999)

À propos des taux d'erreurs

http://aclweb.org/aclwiki/

System name	Short description	Main publication	Software	Extra Data?***	All tokens	Unknown
TnT*	Hidden markov model	Brants (2000)	TnT	No	96.46%	85.86%
MElt	MEMM with external lexical information	Denis and Sagot (2009)	Alpage linguistic workbench	No	96.96%	91.29%
GENiA Tagger**	Maximum entropy cyclic dependency network	Tsuruoka, et al (2005)	GENiA	No	97.05%	Not availa
Averaged Perceptron	Averaged Perception discriminative sequence model	Collins (2002)	Not available	No	97.11%	Not availa
Maxent easiest-first	Maximum entropy bidirectional easiest-first inference	Tsuruoka and Tsujii (2005)	Easiest-first	No	97.15%	Not availa
SVMTool	SVM-based tagger and tagger generator	Giménez and Márquez (2004)	SVMTool	No	97.16%	89.01%
Morče/COMPOST	Averaged Perceptron	Spoustová et al. (2009)	[1]	No	97.23%	Not availa
Stanford Tagger 1.0	Maximum entropy cyclic dependency network	Toutanova et al. (2003)	Stanford Tagger	No	97.24%	89.04%
Stanford Tagger 2.0	Maximum entropy cyclic dependency network	Manning (2011)	Stanford Tagger	No	97.29%	89.70%
Stanford Tagger 2.0	Maximum entropy cyclic dependency network	Manning (2011)	Stanford Tagger	Yes	97.32%	90.79%
LTAG-spinal	Bidirectional perceptron learning	Shen et al. (2007)	LTAG-spinal	No	97.33%	Not availa
Morče/COMPOST	Averaged Perceptron	Spoustová et al. (2009)	[2]	Yes	97.44%	Not availa
SCCN	Semi-supervised condensed nearest neighbor	Søgaard (2011)	SCCN	Yes	97.50%	Not availa

mesuré sur le WSJ

À propos du tagging

- parfois difficile de donner un tag à un mot:
  - mot compressés: *cannot, gonna, wanna*, etc.
  - expressions multi-mots: *pomme de terre, vice et versa*, etc.
  - réelle ambiguité: *The Duchess was entertaining last night*
    adj ou vb?

    (ex. pris de *Part-of-speech Tagging Guidelines for the Penn Treebank Project*)
  - utilisation / mention: *Le mot mot a 3 lettres*

HMM et taggeurs

- Soit w_1^n une séquence de n mots; alors on cherche la séquence de tags de plus forte probabilité:

$$\begin{aligned} \hat{t_1^n} &= \text{argmax}_{t_1^n}\, p(t_1^n|w_1^n) \\ &= \text{argmax}_{t_1^n}\, p(w_1^n|t_1^n) \times p(t_1^n) \end{aligned}$$

- Avec hypothèse d'indépendance + hypothèse markovienne (ordre 1 ici):

$$p(w_1^n|t_1^n) \times p(t_1^n) = \prod_{i=1}^{n} p(w_i|t_i) \times p(t_i|t_{i-1})$$

- D'où:

$$\hat{t_1^n} = \underset{t_1^n}{\text{argmax}} \prod_{i=1}^{n} \underbrace{p(w_i|t_i)}_{\text{émission}} \times \underbrace{p(t_i|t_{i-1})}_{\text{transition}}$$

Entraînement d'un taggeur HMM

À partir d'un corpus d'entraînement étiqueté

- Un état = une étiquette
- estimées MLE (fréquence relative):

$$p(w|t) = \frac{|(w,t)|}{\sum_w |(w,t)|} = \frac{|(w,t)|}{|t|}$$
$$p(t|t') = \frac{|t't|}{\sum_t |t't|} = \frac{|t't|}{|t'|}$$

  - où (w,t) désigne le fait que w est étiqueté par le tag t;
  - et $t't$ représente la séquence de deux tags t' et t.
    note: $p(t|t')$ est simplement un modèle bigramme

- (Merialdo 1994) utilise un corpus annoté de 40 000 phrases
- les taggeurs du RALI ont été entraînés à partir de corpus d'environ 100 000 mots ($\sim$ 5000 phrases par langue)

Problème avec l'estimateur MLE

- une transition légitime peut ne pas avoir été observée dans TRAIN. Sa probabilité devient cependant nulle.
- un mot n'a peut-être pas été étiqueté dans TRAIN avec toutes ses formes possibles.
  - ex: on a peut-être toujours rencontré garde comme un **NomC-masc-sing** alors qu'il peut apparaître comme:
    - **NomC-fem-sing**
    - **verbe** (à différents temps et personnes).
  - Mais $p(\textit{garde}|\text{NomC-fem-sing}) = 0$.
- mots inconnus (Viterbi ne marchera pas forcément)

À propos du décodeur

(Recherche de la séquence optimale)

- Viterbi: $\hat{t}_1^n = \text{argmax}_{t_1^n} p(t_1^n | w_1^n)$
- Critère local: $\hat{t} = \text{argmax}_t p(t | w_1^n)$

- (Merialdo 1994) montre que cela ne fait pas de grande différence:
  - avec viterbi, les erreurs arrivent (potentiellement) en grappes
  - avec l'approche locale, il y a (potentiellement) plus de foyers d'erreur

- Le plus courant est tout de même le décodage global (viterbi). Lire cependant (Johnson 2007).

Gestion des mots inconnus

- **idée 1:** un mot inconnu peut potentiellement être associé à tous les tags *ouverts*:
  - tag ouvert: tous les tags sauf ceux tels que les prépositions ou les articles (dont on connaît tous les représentants).
  - $p(\text{UNK}|t)$ pour tous les tags autorisés $\Longrightarrow$ lissage

- **idée 2:** s'aider des propriétés formelles du mot à étiqueter:
  - Les suffixes comme iques, tions, ments peuvent fournir (en français) des indices
  - Le fait qu'un mot soit en majuscule est également un indicateur (nom propre, acronyme).
  - Par exemple en estimant: $p(\text{UNK}, \text{end=iques}, \text{capital}|t)$ (on fait souvent l'hypothèse d'indépendance de ces traits)

Pourquoi s'arrêter à un taggeur bigramme ?

- Si le corpus d'entraînement est assez grand, on peut calculer les paramètres d'un taggeur trigramme:

$$\hat{t_1^n} = \underset{t_1^n}{\operatorname{argmax}} \prod_{i=1}^{n} p(w_i|t_i) \times p(t_i|t_{i-2}t_{i-1})$$

- permet de désambiguïser plus de choses:
  Ex: l'étiquette à associer à fatigue dans la fatigue dépend de ce qui précède la.

il	la fatigue	$\rightarrow$	Verb
de	la fatigue	$\rightarrow$	NomC

Augmenter l'ordre du modèle

- Pas toujours payant:
  - Ex: pas de dépendance forte entre deux tags séparés par une virgule:
    $p(t|NomC, \text{VIRGULE}) \approx p(t|\text{VIRGULE})$
- Combiner linéairement plusieurs types de modèles (bi- tri-grammes)
- Modèles à mémoire variable:
  - par analyse/correction manuelle: si on repère une erreur systématique de l'étiqueteur sur une séquence particulière alors on augmente la **mémoire** du modèle pour ce cas.
  - par analyse/correction automatique: (Schütze and Singer 1994; Ristad and Thomas 1997a; Ristad and Thomas 1997b).

Pourquoi s'arrêter à un taggeur bigramme ?

Note: pour augmenter la mémoire d'un modèle, il suffit d'ajouter des états:

mot	tag-1	tag-2	mix
BOS	BOS	BOS	BOS
il	PRON	BOS PRON	BOS PRON
a	AUX	PRON AUX	PRON AUX
dit	VB	AUX VB	AUX VB
,	VIRG	VB VIRG	VIRG
que	CONJ	VIRG CONJ	VIRG CONS
		...	

On change seulement l'étiquetage du corpus.

EM permet-il de s'affranchir d'un corpus déjà étiqueté ?

- non d'après (Merialdo 1994; Elworthy 1994)
  - si on possède un corpus d'entraînement (même de taille modeste) dont on se sert pour initialiser les paramètres, alors dès qu'on lance une itération de forward-backward (EM), on dégrade les performances du modèle. Sauf si les corpus de test et d'entraînement sont très différents.
- oui d'après (Johnson 2007)
  - EM est long à converger, sensible au seed
  - bonne performance si l'on réduit le nombre d'états

Approches de base

- approche markovienne
- cas particulier: chunking
- approche transformationnelle

Cas particulier du tagging: le *Chunking*

- **Définition:** Le chunking consiste à découper une phrase en groupes relevant d'une organisation syntaxique.

  [NP He] [VP reckons] [NP the current account deficit] [VP will narrow] [PP to] [NP only # 1.8 billion] [PP in] [NP September].

- Père du chunking: (Abney 1991) qui recherchait des corrélations entre les tags pour identifier des groupes.
- Campagne d'évaluation: **CONLL'2000** (COmputational Natural Language Learning):
  `http://cnts.uia.ac.be/conll2000/chunking/`

Le corpus CONLL

le jeu de C-tags (IOB)

B-tags marquent le début d'un groupe:

- B-NP marque le premier mot d'un groupe nominal (noun phrase);
- B-VP marque le début d'un groupe verbal, etc.

I-tags marquent un mot dans un groupe qui n'est pas le premier mot du groupe.

- I-NP indique qu'un mot est à l'intérieur d'un groupe nominal (d'au moins deux mots)

autres: O marque des mots comme des parenthèses, ou autres signes de ponctuation qui n'appartiennent pas à un groupe.

Au total 22 étiquettes caractérisant les groupes adjectivaux, adverbiaux, verbaux, nominaux, etc. Les deux étiquettes les plus fréquentes sont *I-NP* et *B-NP*, marquant respectivement le milieu d'un groupe nominal, et son début.

Le corpus CONLL

Exemple de phrase annotée

mot	tag	C-tag
He	PRP	B-NP
reckons	VBZ	B-VP
the	DT	B-NP
current	JJ	I-NP
account	NN	I-NP
deficit	NN	I-NP
will	MD	B-VP
narrow	VB	I-VP

mot	tag	C-tag
to	TO	B-PP
only	RB	B-NP
#	#	I-NP
1.8	CD	I-NP
billion	CD	I-NP
in	IN	B-PP
September	NNP	B-NP
.	.	O

Le corpus CONLL

- Environ 3000 mots du corpus de test n'ont pas été vus dans le corpus d'entraînement:

corpus	mots	types	happax
test	49389	8119	55%
train	220663	19123	49%

- Les étiquettes ne sont pas représentées de manière égale:

I-NP	63307	B-ADVP	4227	I-PP	291	I-INTJ	9
B-NP	55081	B-SBAR	2207	I-CONJP	73	I-UCP	6
O	27902	B-ADJP	2060	I-SBAR	70	I-PRT	2
B-VP	21467	I-ADJP	643	B-CONJP	56	B-UCP	2
B-PP	21281	B-PRT	556	B-INTJ	31		
I-VP	12003	I-ADVP	443	B-LST	10		

Chunker = Tagger

- ex: la sortie d'un tagger "normal" constitue l'entrée d'un IOB-taggeur.

$$
\begin{aligned}
& \text{the deficit could narrow} \ldots \\
\longrightarrow\ & \text{DT NN MD VB} \ldots \\
\longrightarrow\ & \text{BOS-DT DT-NN NN-MD MD-VB} \ldots \\
\longrightarrow\ & \text{B-NP I-NP B-VP I-VP} \ldots
\end{aligned}
$$

- variante de cette idée ("Shallow Parsing as Part-of-Speech Tagging"):

$$
\begin{aligned}
w_i &\underset{HMM-1}{\Rightarrow} POS_i \\
(w_i, POS_i) &\underset{HMM-2}{\Rightarrow} IOB_i \\
(POS_i, IOB_i, POS_{i+1}, IOB_{i+1}) &\underset{HMM-3}{\Rightarrow} \hat{IOB}_i
\end{aligned}
$$

Chunker = Tagger

- Si le corpus d'entraînement est suffisamment grand, on peut entraîner directement un taggeur avec le jeu d'étiquettes des IOB-tags

$$p(w|\text{iob-tag}) \text{ et } p(\text{iob-tag}|\text{iob-tag'})$$

  - Exemple (viterbi sur un C-HMM d'ordre 1):
    - mr.(B-NP) speaker(I-NP) ,(O) our(B-NP) government(I-NP) has(B-VP) demonstrated(I-VP) its(B-NP) support(I-NP) for(B-PP) these(B-NP) important(I-NP) principles(I-NP)
    - [NP mr. speaker], [NP our government] [VP has demonstrated] [NP its support] [PP for] [NP these important principles]

- Note: les HMMs ne donnent pas nécessairement les meilleurs résultats.

Chunker & HMM, quelques chiffres

ordre	Transition			Observation		
	A-matrix	nbp	%	B-matrix	nbp	%
HMM-1	$[24 \times 24]$	162	28%	$[24 \times 17259]$	24606	5.9%
HMM-2	$[157 \times 157]$	856	3.4%	$[157 \times 17259]$	38123	1.4%
HMM-3	$[829 \times 829]$	3213	0.4%	$[829 \times 17259]$	59804	0.4%

ordre est l'ordre du modèle HMM considéré (ordre 1 signifie: $p(t|t')$, ordre 2 signifie $p(t|t''t')$, etc.);

nbp est le nombre de paramètres stockés dans la matrice;

% indique le pourcentage "d'occupation" de la matrice (plus il est proche de 0, plus la matrice est creuse).

Performance/temps sans lissage des probabilités

Apprentissage par fréquence relative (pas de lissage)

	training (8936 sent.)			test (2012 sent.)		
	-logp	time	err	-logp	err	nb
1	242.2	6.4u	6%	184.8	10.3%	573
2	227.3	70.2u	3.6%	154.3	8.9%	329
3	212.8	1287u	1.75%	117.6	9.6%	121

-logp $-\log$ de la vraisemblance de l'observation (meilleur si faible)

time u-temps retourné par la commande unix *time* (Pentium-III sous Linux, charge normale)

err pourcentage de mots mal étiquetés

nb nombre de décodages avec une réponse

Approches de base

- approche markovienne
- cas particulier: chunking
- approche transformationnelle

Taggeurs transformationnels (transformation-based taggers)[1]

- **Idée:** transformer une séquence (incorrecte) de tags à l'aide d'une batterie ordonnée de règles transformationnelles qui permettent d'améliorer la séquence.

- Deux composants:
  - patrons des transformations admissibles
  - apprentissage de l'ordonnancement des transformations

- taggeur populaire (open source) (Brill 1992; Brill 1995)

[1] D'après (Manning and Schütze 1999), p. 363

Les patrons du taggeur de Brill

patron = contexte d'application + réécriture ($t_i \longrightarrow t'_i$)

schéma	t_{i-3}	t_{i-2}	t_{i-1}	t_i	t_{i+1}	t_{i+2}	t_{i+3}
1			—	⋆			
2				⋆	—		
3		—	—	⋆			
4				⋆	—	—	
5	—	—	—	⋆			
6				⋆	—	—	—
7			—	⋆	—		
8			—	⋆		—	
9		—		⋆	—		

- ⋆ est le site potentiel de réécriture
- — indique où un trigger peut apparaître
- ligne 7: si un *trigger* (à déterminer) apparaît juste avant t_i, et qu'un autre (à déterminer) apparaît juste après, alors une réécriture (à déterminer) de t_i peut avoir lieu.

Les patrons - (Manning and Schütze 1999), p. 363

réécriture			contexte
NN	$\longrightarrow$	VB	le tag précédant est la prep. TO
VBP	$\longrightarrow$	VB	un modal (MD) est dans les 3 tags qui précèdent
JJR	$\longrightarrow$	RBR	le tag suivant est JJ
VBP	$\longrightarrow$	VB	un des deux mots précédants est *n't*

- la règle 1 dit: ré-étiquette un nom en verbe (à l'infinitif) s'il est précédé de la préposition **to** (contre-exemple: *go to school*).
- la règle 2 s'applique aux verbes ayant la même forme au passé et au présent (ex: *cut*, *put*) et dit qu'en présence d'un modal (max 3 mots avant), on devrait préférer la forme au présent (exemple: *you may cut*).
- la règle 3 transforme un adjectif comparatif (JJR) en un adverbe comparatif (RBR) s'il est suivi directement d'un adjectif (ex: *the more valuable*).
- la règle 4 est proche de la règle 2 pour le cas des négations (*shouldn't* est coupé en deux mots).

Les patrons du taggeur de Brill

- Les triggers mettent en œuvre des étiquettes, des mots ou des traits sur les mots:
  - le mot courant est *w* et le tag qui suit est *t*
  - remplace NN par NNS si le mot courant se termine par s
- Beaucoup de latitude dans les règles que l'on peut apprendre
- c'est aussi un problème ...

Apprentissage des taggeurs transformationnels

In Un corpus taggé C_0
(ex: tag le plus fréquent pour chaque mot)

Out ordonnancement d'un sous-ensemble de règles:

for k := 0 step 1 **do**
$v := \text{argmin}_{v_i} E(v_i(C_k))$
si $(E(C_k) - E(v(C_k)) < \varepsilon)$ **alors** aller à *fin*
$C_{k+1} := v(C_k)$
$\tau_{k+1} := v$
end
fin: séquence ordonnée: $\tau_1, \ldots \tau_k$

- $E(C_k)$: nb. de mots mal taggés dans C à l'itération k.
- $v(C)$: corpus obtenu en appliquant la règle v sur le corpus C; v_i une règle particulière.
- ε spécifie notre tolérance à l'erreur.

- C'est un algorithme vorace (*greedy algorithm*).

Application des règles de ré-écriture

- de la gauche vers la droite.
- immédiate ou retardée (Brill = retardée).
  Soit la règle $A \longrightarrow B$ si A précède

retardé $AAAA \rightarrow ABBB$
(on marque les transformations à effectuer, puis on les fait)

immédiat $AAAA \rightarrow ABAB$

- Brill reporte un taggeur appris de manière non supervisée (sans corpus taggé) avec un taux de 95.6%.
  - utilise l'information des mots non ambigus (qui possèdent un seul tag, selon le dictionnaire)
  - intuition: can dans The can is open sera taggé NN (et non MD), si dans le contexte "ART — VB", les mots non ambigus sont majoritairement étiquetés NN.

Quelques lectures sur le tagging

- (Brant 2000) petits trucs (habituellement non documentés) qui influent sur la performance. Un tagger avec un très bon rapport performance/vitesse ($\sim$ 100 000 tokens/sec. sur un Pentium 500)

  `http://www.coli.uni-sb.de/~thorsten/tnt`
- (Giménez and Màrquez 2003) pour une comparaison plus récente de TnT *vs* SVMs.
- Approche maxent (Ratnaparkhi 1996) et (Toutanova and Manning 2000)
- (Toutanova, Klein, Manning, and Singer 2003) pour un modèle bidirectionnel: (gauche-droite et droite-gauche)
- Approche *memory-based* (Daelemans, Zavrel, P., and Gillis 1996)
- (Collobert, Weston, Bottou, Karlen, Kavukcuoglu, and Kuksa 2011) pour des performances à l'état de l'art avec des réseaux de neurones
- Une revue intelligente de l'état de l'art et de ses limites (Manning 2011)

Références I


Abney, Steven (1991). *Parsing By Chunks*. Robert Berwick and Steven Abney and Carol Tenny, "Principle-Based Parsing", Kluwer Academic.


Adda, Gilles, Joseph Mariani, Patrick Paroubek, Martin Rajman, and Josette Lecomte (1999). "The GRACE evaluation for POS tagging for French language". In: *Cahiers/Langues*. Vol. Vol. 2, Issue 2. http://www.john-libbey-eurotext.fr/en/revues/lan/index.htm.


Brant, T. (2000). "TnT - A statistical Part-of-Speech Tagger". In: *ANLP*, Seattle, WA.


Brill, Eric (1992). "A simple rule-based part-of-speech tagger". In: *Proceedings of ANLP-92, 3rd Conference on Applied Natural Language Processing*. Trento, IT, pp. 152–155.


— (1995). "Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging". In: *Computational Linguistics* 21.4, pp. 543–565.


Charniak, Eugene (1993). *Statistical Language Learning*. MIT Press.


Collobert, R., J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa (2011). "Natural Language Processing (Almost) from Scratch". In: *Journal of Machine Learning Research* 12, pp. 2493–2537.

Références II


Daelemans, W., J. Zavrel, Berck. P., and S. Gillis (1996). “MBT: A memory-based part of speech tagger generator”. In: *Proc. of Fourth Workshop on Very Large Corpora, ACL SIGDA*, pp. 14–27.


Elworthy, David (1994). “Does Baum-Welch reestimation help taggers?” In: *In Proceedings of the 4th ACL Conference on Applied Natural Language Processing (ANLP’94)*. Stuttgart, Germany.


Giménez, J. and L. Màrquez (2003). “Fast and Accurate Part-of-Speech Tagging: The SVM Approach Revisited”. In: *RANLP*. Borovets, Bulgaria.


Johnson, Mark (2007). “Why Doesn’t EM Find Good HMM POS-Taggers?” In: *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Prague, Czech Republic, pp. 296–305.


Manning, Christopher D. (2011). “Part-of-speech tagging from 97% to 100%: is it time for some linguistics?” In: *Proceedings of the 12th international conference on Computational linguistics and intelligent text processing - Volume Part I*. CICLing’11. Tokyo, Japan, pp. 171–189.


Manning, Christopher D. and Hinrich Schütze (1999). *Foundations of Statistical Natural Language Processing*. MIT Press.

Références III


Merialdo, Bernard (1994). “Tagging English text with a probabilistic model”. In: *Computational Linguistics* 20(2), pp. 155–172.


Osborne, Miles. “Shallow Parsing as Part-of-Speech Tagging”. In:


Ratnaparkhi, A. (1996). “A maximum entropy model for part-of-speech tagging”. In: *EMNLP*. Philadelphia, PA.


Ristad, E. and R. Thomas (1997a). *Hierarchical Non-Emitting Markov models*. Tech. rep. CS-TR-544-97. Department of Computer Science, Princeton University.


Ristad, E. S. and R. G. Thomas (1997b). “Nonuniform Markov Models”. In: *Proc. ICASSP ’97*. Munich, Germany, pp. 791–794.


Schütze, Hinrich and Yoram Singer (1994). “Part-of-Speech tagging using a variable memory Markov model”. In:


Toutanova, K., D. Klein, C. D. Manning, and Y. Singer (2003). “Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network”. In: *NAACL*.


Toutanova, K. and C. Manning (2000). “Enriching the Knowledge Sources used in a Maximum Entropy Part-of-Speech tagger”. In: *Proceedings of the 2000 Joint SIGDAT Conference EMNLP/VLC*, pp. 63–71.