

PosTagger et Parseur Talismane

« Amélioration et évaluation des modèles »

Assaf Urieli

-

Jean-Philippe Fauconnier

CLLE-ERSS - UMR 5263

Axe TAL

Université Toulouse II - Le Mirail

20 juin 2012



PosTagger et Parseur Talismane

Introduction

Talismane
Caractéristiques
Architecture

Configuration

Chaîne de
traitement
Syntaxe traits
Format d'entrée
Format de sortie

PosTagger

Module PosTag
Mise au Point
Résultats

Parseur

Module Parsing
Caractéristiques
Résultats

Perspectives

Références

Références

① Introduction

② Configuration

③ PosTagger

④ Parseur

⑤ Perspectives

1. Introduction

PosTagger et
Parseur
Talismane

Introduction

Talismane
Caractéristiques
Architecture

Configuration

Chaîne de
traitement
Syntaxe traits
Format d'entrée
Format de sortie

PosTagger

Module PosTag
Mise au Point
Résultats

Parseur

Module Parsing
Caractéristiques
Résultats

Perspectives

Références

Références

① Introduction

Talismane

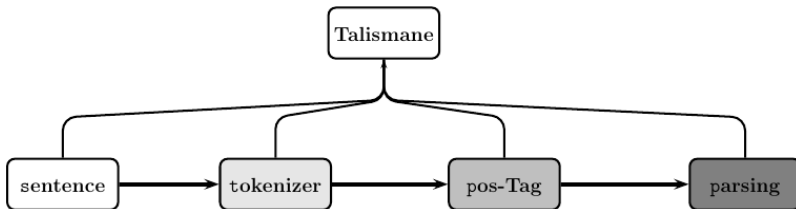
Caractéristiques

Architecture

Talismane :

Traitement **A**utomatique des **L**angues par **I**nférences
 Statistiques **M**oyennant l'**A**nnnotation de **N**ombreux **E**xemples

4 modules consécutifs :



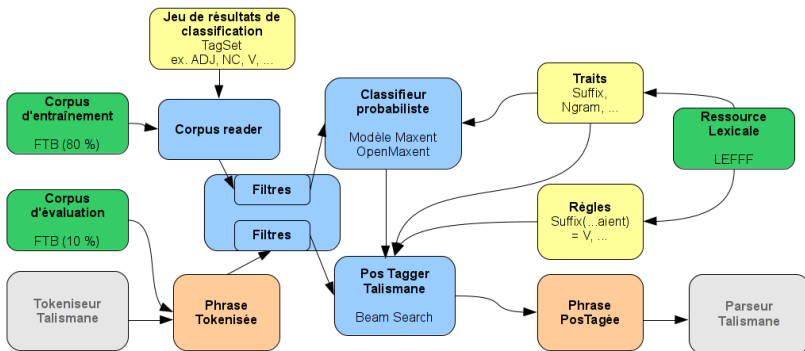
Caractéristiques :

- En Java $\Rightarrow$ Portabilité
- Libre - licence AGPL $\Rightarrow$ Diffusion
- Utilisation d'interfaces

Faciliter de remplacer :

- Classifieur
 - Corpus
 - Lexique
- Facilement configurable
 - Entrée/Sorties
 - TagSet
 - Traits
 - Règles
 - ...

Architecture :



2. Configuration

PosTagger et Parseur Talismane

Introduction

Talismane
Caractéristiques
Architecture

Configuration

Chaîne de
traitement
Syntaxe traits
Format d'entrée
Format de sortie

PosTagger

Module PosTag
Mise au Point
Résultats

Parseur

Module Parsing
Caractéristiques
Résultats

Perspectives

Références

Références

② Configuration

Chaîne de traitement

Syntaxe traits

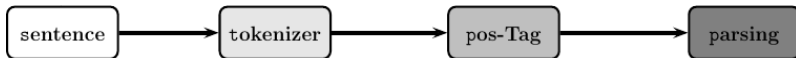
Format d'entrée

Format de sortie

Chaîne de traitement :

- Quatre modules à exécuter en séquence
- Choix de module début et module fin

```
java talismane_fr_0.0.4.jar startModule=tokenise  
endModule=PosTag
```



Syntaxe générique de traits :

- Opérateurs : &, |, ==, !=, +, -, /, *, ...
- Fonctions génériques.

ex : IfThenElse, Truncate, Round, Inverse, ...

- Fonctions spécifiques par module.

ex : Parseur = DependencyLabel, DependencyCountIf

ex : Vérifie si le verbe en haut de la pile a déjà un sujet.

```
(PosTag(Stack[0])=="V" | PosTag(Stack[0])=="VS")) & (  
    DependentCountIf(Stack[0], DependencyLabel()=="subj")  
    > 0)
```

Format d'entrée configurable :

ex : Format d'entrée pour le parseur.

```
. * \ t TOKEN \ t . * \ t POSTAG \ t . * \ t . * \ t
```

ex : Fichier d'entrée

```
1 Je je CLS cl 1s
```

Format de sortie configurable avec FreeMarker :

```
[#ftl]
[#list sentence.units as unit]
${unit.lexicalEntry.lemma! "_"}  ${unit.label! "_"}
${(unit.governor.lexicalEntry.lemma)! "_"}
[/#list]
```

Résultat :

manger	SUJ	il
manger	AUX_TPS	avoir
pomme	DET	les
manger	OBJ	pomme
pomme	DE	terre

3. PosTagger

PosTagger et Parseur Talismane

Introduction

Talismane
Caractéristiques
Architecture

Configuration

Chaîne de
traitement
Syntaxe traits
Format d'entrée
Format de sortie

PosTagger

Module PosTag
Mise au Point
Résultats

Parseur

Module Parsing
Caractéristiques
Résultats

Perspectives

Références

Références

③ PosTagger

Module PosTag

Entrée/Sortie
Philosophie Unix
Architecture PosTagger

Mise au Point

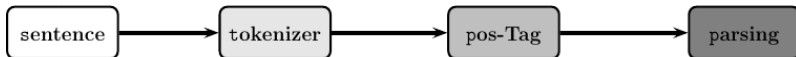
Corpus d'erreurs
Traits
Règles
Code-Source

Résultats

Résultats
Erreurs FTB
Benchmark

Le Module PosTag :

- ❶ Propose un framework pour le PosTagging
- ❷ Potentiellement adaptable à chaque langue
- ❸ Écrit en Java et licence libre $\Rightarrow$ Portabilité, Diffusion
- ❹ Intégration dans la suite Talismane :
 - Classifieur probabiliste
 - Syntaxe des traits
 - Architecture ouverte



PosTagger pour le Français :

Classifieur : Apache OpenNLP MaxEnt

Corpus : French Treebank

Lexique : Lefff

TagSet : Crabbé et Candito

Traits/Règles : Adapté au français

PosTagger et Parseur Talismane

Introduction

Talismane
Caractéristiques
Architecture

Configuration

Chaîne de traitement
Syntaxe traits
Format d'entrée
Format de sortie

PosTagger

Module PosTag
Mise au Point
Résultats

Parseur

Module Parsing
Caractéristiques
Résultats

Perspectives

Références

Références

Entrée :

Une phrase délimitée et découpée en tokens

```
Les      les
poules   poule
du       de
-        le
couvent  couvent
couvent  couver
.
```

Sortie :

La phrase adjointe de posTags + infos morpho-syntaxiques (sous-spécifiées)

```
0   Les      les      DET      det      p
1   poules   poule   NC       nc       fp
2   du       de       P        prep     null
3   -        le       DET      det      ms
4   couvent  couvent  NC       nc       ms
5   couvent  couver   V        v        PS3p
6   .        .        PONCT   poncts   null
```

PosTagger et Parseur Talismane

Introduction

Talismane
Caractéristiques
Architecture

Configuration

Chaîne de
traitement
Syntaxe traits
Format d'entrée
Format de sortie

PosTagger

Module PosTag
Mise au Point
Résultats

Parseur

Module Parsing
Caractéristiques
Résultats

Perspectives

Références

Références

Proche philosophie Unix :

- Brique élémentaire qui fournit un type de résultat
- Pas d'interface graphique
- Passage de paramètres
- Gestion des flux `stdin/stdout`
- Traitement avec les outils de base Unix

```
echo "Les poules du couvent couvent." | java -Xmx1024M
-jar talismane-fr.jar startModule=sentence
endModule=postag encoding=UTF-8 | cut -f2,4
```

Les	DET
poules	NC
du	P
-	DET
couvent	NC
couvent	V
.	PONCT

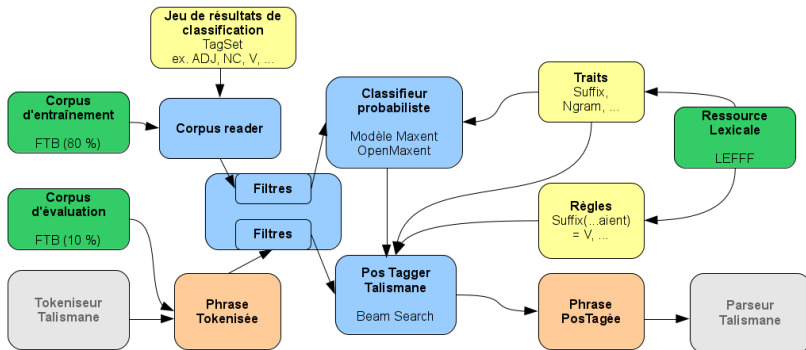
Architecture du PosTagger :

Architecture **générique** + *implémentation* pour le français

Configuration : TagSet, Traits, Règles

Ressources : Corpus train/eval, Lexique formes fléchies

Outils : Corpus reader, Classifieur, PosTagger



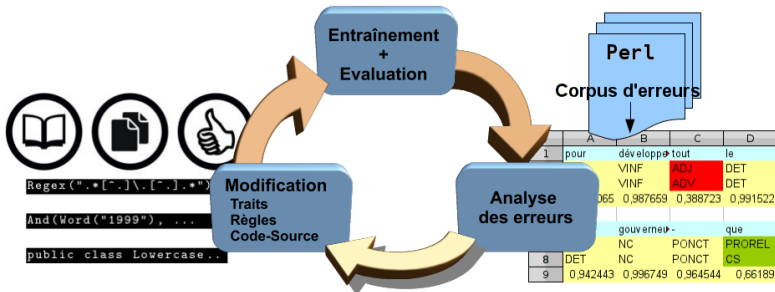
Mise au Point : Modèle itératif :

Entraînement + Évaluation : Puissance de calcul

Analyse des erreurs : Chaîne Perl ⇒ corpus d'erreurs

Modifications : Travail d'équipe

- Développement filtres (Majuscules, Nombres, ...)
- Nouveaux types de traits (Regex, Opérateurs, ...)
- ...



Création d'un corpus d'erreurs :

Directement créé à partir de l'évaluation/analyse du modèle

	A	B	C	D	E	F	G	H	I	J
1	,	disant	tout	le	bien	qu'	il	pense	de	Chirac
2	PONCT	VPR	ADJ	DET	NC	PROREL	CLS	V	P	NPP
3	PONCT	VPR	ADV	DET	NC	CS	CLS	V	P	NPP
4	0,98482554	0,69650037	0,33129568	0,88405562	0,50837622	0,86703925	0,99568915	0,99306606	0,95939199	0,99174045
5										
6	à	la	barbe	folle	et	aux	[[aux]]	yeux	d'	albinos
7	P	DET	NC	ADJ	CC	P	DET	NC	P	NC
8	P	DET	NC	ADJ	CC	P	DET	NC	P	NC
9	0,96056683	0,99804305	0,97596965	0,98390827	0,97462868	0,99076044	0,99912578	0,99433406	0,96189036	0,8897699
10										
11	d'	encours	sur	le	livret	A	,	celui-là	même	qui
12	P	NC	P	DET	NC	ADJ	PONCT	PRO	ADJ	PROREL
13	P	NC	P	DET	NC	NPP	PONCT	PRO	ADV	PROREL
14	0,98847472	0,98635685	0,98943008	0,99858158	0,99646319	0,10600031	0,98794737	0,79192399	0,5628285	0,96096957
15										
16	déficit	de	trésorerie	de	quelque	999	milliards	de	francs	pendant
17	NC	P	NC	P	ADJ	DET	NC	P	NC	P
18	NC	P	NC	P	ADV	DET	NC	P	NC	P
19	0,99596418	0,994731	0,99864341	0,99761669	0,67824451	0,88439491	0,99645937	0,9912195	0,99872643	0,99101793
20										
21	une	hausse	d'	un	point	de	cette	cotisation	rapporterait	quelque
22	DET	NC	P	DET	NC	P	DET	NC	V	ADJ
23	DET	NC	P	DET	NC	P	DET	NC	V	ADV
24	0,991685	0,996588	0,99907	0,987447	0,992906	0,997504	0,995000	0,999406	0,996859	0,686188

Description :

Attribut dont un token peut disposer

- Permet de prendre en compte les **régularités** de la langue.
- Informe sur le **PosTag** (NC, ADJ, ...) d'un Token
Ex : séquence : *det, adj, ?* $\Rightarrow \{NC, ADV, \dots\}$
Ex : terminaison : *-ement* $\Rightarrow \{NC, ADV\}$
- Le Classifieur apprend au travers d'**événements**.
événement = paire(série de traits, PosTag)
- Retourne valeur booléenne, numérique ou suite de caractères
- Les traits sont utilisés à l'**entraînement** et **évaluation/analyse** du modèle

Le Module PosTag pour le français

- Usage du Lefff
- Seuil à 5 $\Rightarrow$ *Problème du « tu »*
- Deux approches pour les traits
 - Traits génériques
 - Traits spécifiques

Traits génériques :

Traits qui s'appliquent à n'importe quelle forme lexicale

- `WordForm()` : tu (CLS 95%, ET 5 %)
- `Ngram(3)` : DET - ADJ - ? (NC 80%, ADJ %15, PONCT 5%)
- `NLetterSuffix(IndexRange(2,5))` : Suffixe -es / -ent / -ment / -aient
- `LexiconPosTag("V")` : Est-ce que le Lefff renvoie "Verbe" pour ce token ?
- `UnknownWord()` : Est-ce que le mot est connu dans le Lefff ?
- `Regex (" .+-.+ ")` : Est-ce que le mot contient un tiret ?

Traits spécifiques :

Traits qui s'appliquent à une forme lexicale spécifique

- `Word("999") & Offset(1, LexiconPosTag("NC"))` :
Les 3 chats vs 3 chiens

Archétype du problème de l'approche statistique :

Corpus training : 800 000 tokens mais

- 3 « tu » CLS
- 2 « Tu » CLS
- 1 « Tu » ET

⇒ Trop bas pour le seuil

⇒ Résultat du Lefff : CLS ou VPP

Phrase	PosTag
Il m'a affirmé, au contraire : tu fais le mieux qu'il est possible (...)	CLS
(...) venu me dire : tu t'y prends mal, ça ne va pas du tout, (...)	CLS
Tu parles !	CLS
(...) cassettes ; chaque fois que je l'ouvre, tu fais pareil, (...)	CLS
(...) de culpabilité : " Tu perds ton temps devant des âneries !"	CLS
(...) Brahms, de Verdi, l'air " Tu che la vanità" de Don Carlos (...)	ET

Solutions :

- Baisser le seuil arbitrairement
- Agrandir artificiellement le corpus
- Trouver un trait qui rende statistiquement suffisant le « tu »

```
PrevPosTagIs("PONCT") & LexiconPosTag("CLS")
```

PosTagger et Parseur Talismane

Introduction

Talismane
Caractéristiques
Architecture

Configuration

Chaîne de
traitement
Syntaxe traits
Format d'entrée
Format de sortie

PosTagger

Module PosTag
Mise au Point
Résultats

Parseur

Module Parsing
Caractéristiques
Résultats

Perspectives

Références

Références

Description :

Classification conditionnelle d'un token

- Classifie automatiquement un token
- Tout trait booléen peut être transformé en règle
- Stoppe la classification motivée statistiquement
- L'ordre des règles a de l'importance
- Les règles sont utilisées à **l'évaluation/analyse** du modèle
Pas dans l'entraînement.

Le Module PosTag pour le français

- Usage du Lefff
- Utile pour « étendre le corpus »
- mais problème maintenance / robustesse.
- Deux approches pour les règles
 - Contraintes
 - Cas particuliers

Contraintes :

Gestion de classe fermée, classe ouverte

- `!PONCT Not(LexiconPosTag(" PONCT" ))`
On ne le classe pas comme PONCT si le token n'est pas listé comme PONCT dans le Lefff.
- `!PRO Not(LexiconPosTag(" PRO" ))`
- `!NC HasClosedClassesOnly()`
On ne le classe pas comme NC si le token appartient **uniquement** à des classes fermées.
- `!ADJ HasClosedClassesOnly()`
Ex : « le » existe dans le Lefff : CLO (fermé) & DET (fermé).

Cas particuliers :

Contraindre un token pour l'appartenance à une catégorie

- `ADJ PrevPosTagIs(-1," DET" ) & Word(" 999" )`
ex : Les 3 chats
- `ADJ PrevPosTagIs(-1," DET" ) & Word(" deux" ," trois" ," quatre" ," cinq" ,...)`
ex : Les deux/trois/quatre/ ... chats

Code-Source :

Apports dans la mise au point du PosTagger pour le français

- **Correction de Filtres**
Traiter les majuscules après une ponctuation
- **Ajout de Filtres**
Traiter les séquences de mots tout en majuscules Remet en minuscules si ces mots sont connus dans le Lefff SAFARI SA est une entreprise.
LE PRESIDENT se rend au Maroc.
- **Ajout de fonctions dans les traits**
Nomination des traits, Allègement syntaxe, Fonction Regex, ...

```
IsDayOfMonth Word("31") & AndRange(Word("31","au",
",","et"), 1, ForwardLookup(IsMonthName())
-1)
```


PosTagger et Parseur Talismane

Introduction

Talismane
Caractéristiques
Architecture

Configuration

Chaîne de
traitement
Syntaxe traits
Format d'entrée
Format de sortie

PosTagger

Module PosTag
Mise au Point
Résultats

Parseur

Module Parsing
Caractéristiques
Résultats

Perspectives

Références

Références

Présentations des résultats :

Pour l'instant, 4 résultats à comparer dans le corpus DEV

- Baseline
- Filtres
- Filtres + Traits
- Filtres + Traits + Règles

Pour l'instant, 3 résultats à comparer avec le corpus EVAL

- DEV + EVAL
- DEV + Règles
- EVAL + Règles

Erreurs dans le FTB :

① Définition d'une erreur

- Soit il s'agit d'une erreur de transcription dans le corpus.
- Soit le PosTag attribué par l'annotateur ne correspond pas au *Guide Morpho-Syntaxique*.

après	la	génération	du	[[du]]	Déluge	,	il	est	devenu	carnivore
P	DET	NC	P	DET	NC	PONCT	CLS	V	VPP	ADJ
P	DET	NC	P	DET	NPP	PONCT	CLS	V	VPP	ADJ
0,965013	0,990000	0,998718	0,996315	0,998666	0,664562	0,999045	0,952042	0,998047	0,781194	0,546465
soit	dit	en	passant	,	les	Bantous	gèrent	aussi	la	violence
VS	VPP	P	VPR	PONCT	DET	NC	V	ADV	DET	NC
CC	VPP	P	VPR	PONCT	DET	NPP	V	ADV	DET	NC
0,335205	0,702903	0,983327	0,849589	0,983896	0,986243	0,561309	0,996770	0,868962	0,943157	0,992693
quatre	d'	entre	eux	seront	diffusés	sur	Canal+	dans	Mouche	noire
PRO	P	P	PRO	V	VPP	P	NPP	P	NC	ADJ
PRO	P	P	PRO	V	VPP	P	NPP	P	NPP	ADJ
0,936509	0,952188	0,782941	0,983644	0,996290	0,967531	0,958533	0,980246	0,988857	0,857266	0,965264

② Correction d'erreurs ?

- Utilité pratique
- Utilité scientifique
- Problème d'évaluation face aux autres outils

PosTagger et Parseur Talismane

Introduction

- Talismane
- Caractéristiques
- Architecture

Configuration

- Chaîne de traitement
- Syntaxe traits
- Format d'entrée
- Format de sortie

PosTagger

- Module PosTag
- Mise au Point
- Résultats

Parseur

- Module Parsing
- Caractéristiques
- Résultats

Perspectives

Références

- Références

Benchmark :

Graphe GnuPlot :

script bash : `time java -Xmx1024M -jar talismane-fr ; file1`

PosTagger et Parseur Talismane

Introduction

- Talismane
- Caractéristiques
- Architecture

Configuration

- Chaîne de traitement
- Syntaxe traits
- Format d'entrée
- Format de sortie

PosTagger

- Module PosTag
- Mise au Point
- Résultats

Parseur

- Module Parsing
- Caractéristiques
- Résultats

Perspectives

Références

- Références

④ Parseur

- Module Parsing

 - Entrées-Sorties

 - Architecture

- Caractéristiques

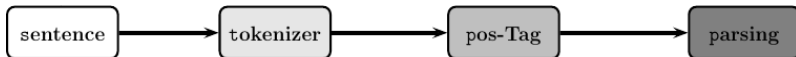
 - Algorithme

 - Traits

- Résultats

Le Module de Parsing :

- ❶ Propose un framework pour le parsing (analyse syntaxique)
- ❷ Potentiellement adaptable à chaque langue
- ❸ Écrit en Java et licence libre $\Rightarrow$ Portabilité, Diffusion
- ❹ Intégration dans la suite Talismane :
 - Classifieur probabiliste
 - Syntaxe des traits
 - Architecture ouverte



Parseur pour le Français :

Classifieur : Apache OpenNLP MaxEnt

Corpus : French Treebank en dépendances (Crabbé et Candito)

Lexique : Lefff

TagSet : Crabbé et Candito

Traits/Règles : Adapté au français

PosTagger et Parseur Talismane

Introduction

Talismane
Caractéristiques
Architecture

Configuration

Chaîne de traitement
Syntaxe traits
Format d'entrée
Format de sortie

PosTagger

Module PosTag
Mise au Point
Résultats

Parseur

Module Parsing
Caractéristiques
Résultats

Perspectives

Références

Références

Entrée :

Une phrase postagagée

1	Les	les	DET	det	p
2	poules	poule	NC	nc	fp
3	du	de	P	prep	null
4	_	le	DET	det	ms
5	couvent	couvent	NC	nc	ms
6	couvent	couver	V	v	PS3p
7	.	.	PONCT	poncts	null

Sortie :

La même phrase en arbre syntaxique de dépendances

1	Les	les	DET	det	p	2	det
2	poules	poule	NC	nc	fp	6	suj
3	du	de	P	prep	null	2	dep
4	_	le	DET	det	ms	5	det
5	couvent	couvent	NC	nc	ms	3	obj
6	couvent	couver	V	v	PS3p	0	root
7	.	.	PONCT	poncts	null	6	ponct

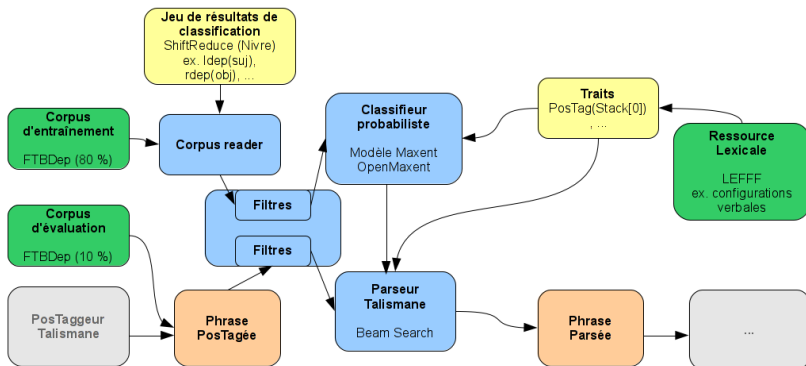
Architecture du Parseur :

Architecture **générique** + *implémentation* pour le français

Configuration : Jeux de résultats de classification, Traits

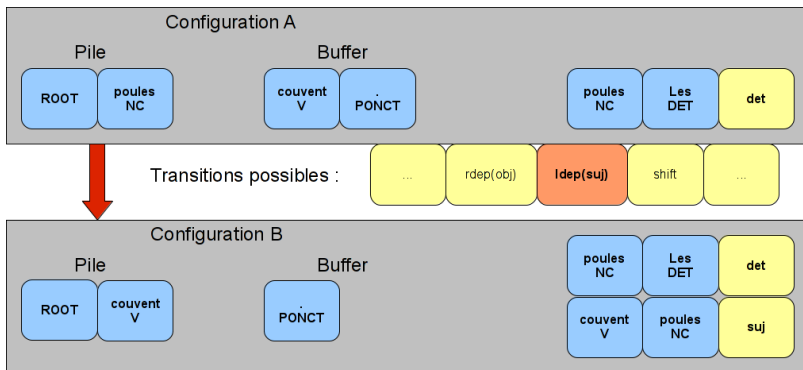
Ressources : Corpus train/eval, Lexique formes fléchies

Outils : Corpus reader, Classifieur, Parseur



Transformer le parsing en un problème de classification :

Shift-reduce transition based parsing (Nivre 2008)



Exemples de traits utilisés pour le Parsing :

- `PosTag(Stack[0])`
- `PosTag(Buffer[0])`
- `PosTagPair Concat(PosTag(Stack[0]),PosTag(Buffer[0]))`
- **GenderMatch**
`Concat(PosTag(Stack[0]),Gender(Stack[0]),PosTag(Buffer[0]),Gender(Buffer[0]))`
- **StackHasSubject**
`(PosTag(Stack[0])=="V" — PosTag(Stack[0])=="VS") &
(DependentCountIf(Stack[0],DependencyLabel()=="subj")>0)`
- **NormalisedDistance**
`Normalise(Distance(Stack[0],Buffer[0]),0.0,10.0)`
- **SingleComma**
`BetweenCountIf(Stack[0],Buffer[0],LexicalForm()==" , ")==1`
- **MultipleCommas**
`BetweenCountIf(Stack[0],Buffer[0],LexicalForm()==" , ")>1`

PosTagger et Parseur Talismane

Introduction

Talismane
Caractéristiques
Architecture

Configuration

Chaîne de
traitement
Syntaxe traits
Format d'entrée
Format de sortie

PosTagger

Module PosTag
Mise au Point
Résultats

Parseur

Module Parsing
Caractéristiques
Résultats

Perspectives

Références

Références

Résultats du Parseur :

Parseur	accuracy (hors ponct.)
Crabbé et Candito	87.3%
Talismane : Baseline	79.5%
Talismane : Après mise au point initiale	86.0%

PosTagger et Parseur Talismane

Introduction

- Talismane
- Caractéristiques
- Architecture

Configuration

- Chaîne de traitement
- Syntaxe traits
- Format d'entrée
- Format de sortie

PosTagger

- Module PosTag
- Mise au Point
- Résultats

Parseur

- Module Parsing
- Caractéristiques
- Résultats

Perspectives

Références

- Références

⑤ Perspectives

Perspectives :

- Publier le code
- Publier la documentation en HTML
- Préparer l'évaluation face à d'autres systèmes
- Essai sur corpus autres que journalistiques et/ou « sales »
- Expérimenter avec d'autres TagSets
- Améliorer en amont la gestion des patterns complexes
URL, Entités Nommées, ...
- Mise au point du Parseur
- Mise au point des sorties du Parseur
triplets (mémoire distributionnelle), liste des SN avec fréquences

PosTagger et Parseur Talismane

Introduction

Talismane
Caractéristiques
Architecture

Configuration

Chaîne de
traitement
Syntaxe traits
Format d'entrée
Format de sortie

PosTagger

Module PosTag
Mise au Point
Résultats

Parseur

Module Parsing
Caractéristiques
Résultats

Perspectives

Références

Références

⑥ Références Références

PosTagger et Parseur Talismane

Introduction

Talismane
Caractéristiques
Architecture

Configuration

Chaîne de
traitement
Syntaxe traits
Format d'entrée
Format de sortie

PosTagger

Module PosTag
Mise au Point
Résultats

Parseur

Module Parsing
Caractéristiques
Résultats

Perspectives

Références

Références

Références bibliographiques :

- Abeille, ...
- Candito et Crabbé, ...