Cross Parser Evaluation and Tagset Variation : a French Treebanks Study

Djamé Seddah◇, Marie Candito* et Benoît Crabbé*

◇ Université Paris-Sorbonne
LaLIC et Inria (Alpage)

* Université Paris 7
Ufrl et Inria (Alpage)

December 09, Feast, Saarland University

Probabilistic Parsing and the Case of French

Motivations

Until recently, French language has not attracted much attention from the probabilistic parsing community

- 2003, Initial availability of the Paris 7 French Treebank
- 2004, Dybro-Johansen (with Alexis Nasr) : TAG extraction
- 2004-2005, Arun & Keller : Collins' model 2 adaptation

Hypothesis for this lack of interest

- French parsing community was focusing on wide coverage symbolic parsing (ie. development of various wide coverage Metagrammars -XMG, FRMG-)
- Treebank a bit challenging to manipulate (lack of functional annotations, inconsistencies, XML errors...)

Probabilistic Parsing and the Case of French

Motivations

Until recently, French language has not attracted much attention from the probabilistic parsing community

- 2003, Initial availability of the Paris 7 French Treebank
- 2004, Dybro-Johansen (with Alexis Nasr) : TAG extraction
- 2004-2005, Arun & Keller : Collins' model 2 adaptation

Then, simultaneous availability of new annotated data

- 2007, Schluter & van Genabith : Collins' model 2 adaptation on (already) a new French treebank
- 2008, Crabbé & Candito : adaptation of the Berkeley parser
- 2008, Schluter & van Genabith : Wide coverage LFG parsing

Probabilistic Parsing and the Case of French

As things seem to be moving...

Brutal questions are raised :

⇒ *So far, we only have reported results for the Collins' models and the Berkeley parser*

- How would behave other lexicalized models ?
- Which treebank offers the best performance ?
- **What is the state of the art for French anyway ?**

Underlying question

Does lexicalization matter for French ?

- (Arun & Keller, 05) : *"lexicalization is useful but treebanks' flatness limits its impact"*
  Treebank : very early version (Ftb-v0)
- (Schluter & van Genabith, 07) : *" Indeed, it does, a bit, but not as much as a consistant treebank built with parsing and grammar induction consideration in mind"*
  Treebank : Modified version of the FTB (Mft)
- (Candito & al, 09) : *"An unlexicalized PCFG-LA parser provides very good performance and outperforms lexicalized models such as Collins' models based ones "*
  Treebank : Corrected version of the FTB and a specific tagset,Ftb-cc

Probabilistic Parsing and the Case of French

- Fact

  Results on 3 different treebanks released at 3 different moments
- Issue

  How to evaluate the impact of different parsing models on corpus with different annotation schema without any stable gold standard ?
- Solution

  We settled for exhaustivity

Goal : a complete study

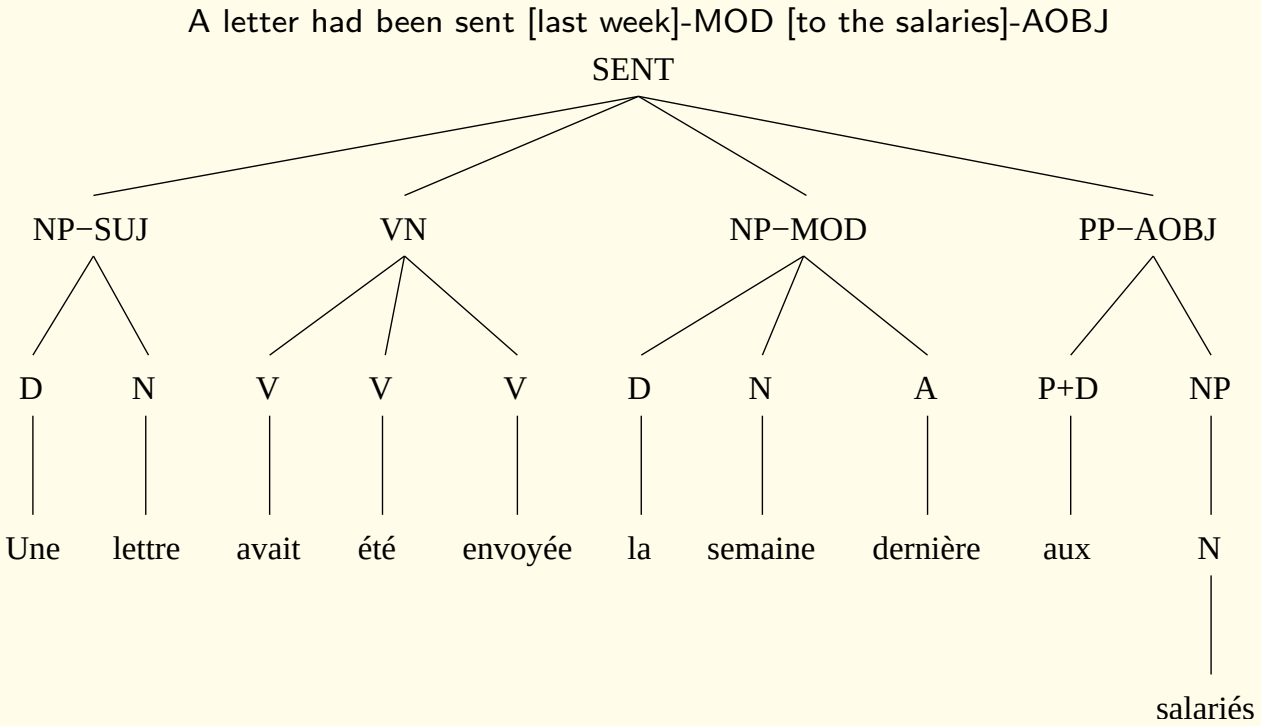
- **Evaluate the performance of the main lexicalized models**
  - Charniak's model
  - Collins (Model 1,2) and Bikel (model X)
  - Chiang's STIG model (pure and spinal)
- **w.r.t an unlexicalized model**
  - PCFG with Latent Annotation (Berkeley parser)
- **On the French Treebanks**
  - the Paris 7 Treebank (Ftb, Abeillé et al, 2003)
  - the Modified French Treebank (Mft, (Schluter et van Genabith, 2007)
- **by evaluating the influence of tagset**
  - via constituency metrics (Parseval's Labeled Brackets)
  - via untyped dependency metrics (Lin, 95)

Outline

1. **Data Set : FTB and MFT**
2. Experimental Protocol
3. Cross Parsing experiments
4. Treebanks Intersection *(ongoing work)*
5. Discussion on lexicalized models

Data Set : The FTB

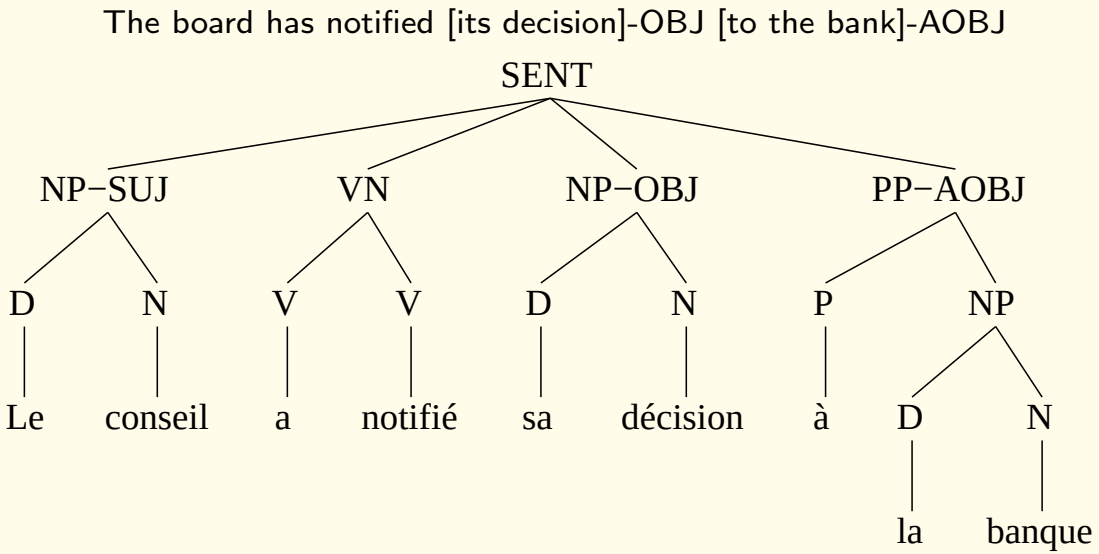
- Flat Annotation Scheme : (Rejects X' syntax)
- Distinction between arguments and adjuncts not configurational.



- Only possible with functional annotations

Data Set : The FTB

- Flat Annotation Scheme : (Rejects X' syntax)
- Distinction between arguments and adjuncts not configurational.



- Only possible with functional annotations

Data Set: the "Modified French Treebank" (MFT)

- The "Modified French Treebank", (Schluter and van Genabith 07,08) : Version based on subset of the FTB (2004)
- Main goal : Optimize grammar induction and LFG parsing
  - Increased stratification (eg. VP node added on some coordinate structure)
  - Modified coordination scheme
  - Extensive modifications of the annotation scheme to ease grammar induction
  - Exhaustive manual corrections and phase of error mining

MFT vs FTB : Figures and examples

properties	Ftb	Mft
# of sentences	12351	4739
Average sent. length	27.48	28.38
PCFG size (without term. prod.)	14874	6944
PCFG avg. node branching	6.42	5.66
# of NT symbols	13	39
# of POS tags	15	27

Table: Treebanks Properties

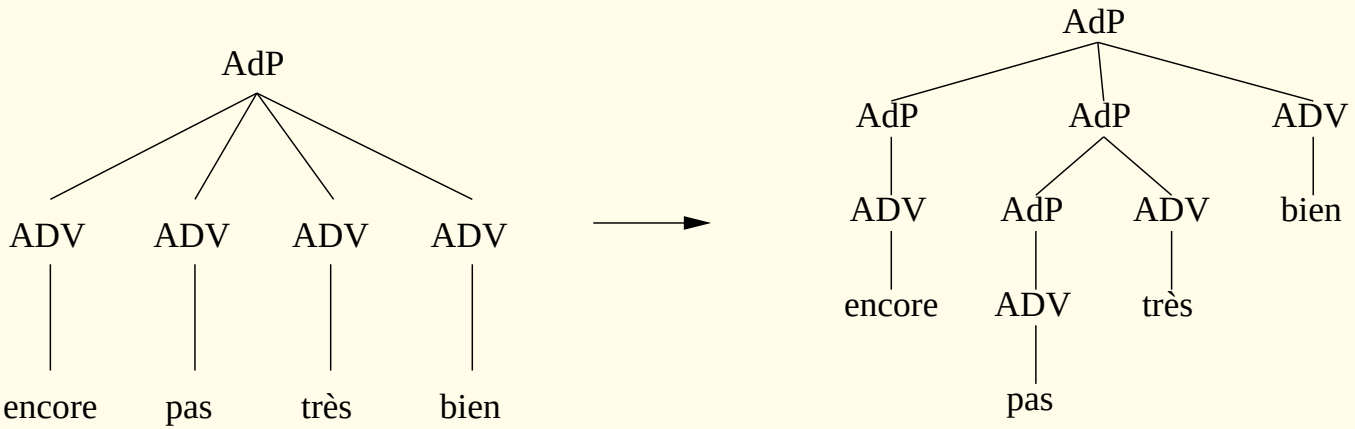


Figure: Increased stratification : from FTB to MFT (Schluter et van Genabith, 2007)

MFT vs FTB: Annotation Scheme

	Ftb (base categories)	Mft (native)
POS tags	A ADV C CL D ET I N P P+D P+PRO PONCT PREF PRO V	A A_card ADV ADV_int ADVne A_int CC CL C_S D D_card ET I N N_card P P+D PONCT P+PRO_rel PREF PRO PRO_card PRO_int PRO_rel V_finite V_inf V_part
NT labels	AP AdP COORD NP PP SENT Sint Srel Ssub VN VPinf VPpart	AdP AdP_int AP AP_int COORD_XP COORD_UC COORD_unary NC NP NP_int NP_rel PP PP_int PP_rel SENT Sint Srel Ssub VN_finite VN_inf VN_part VP VPinf VPpart VPpart_rel

Table: Ftb's and Mft's annotation schemes

- Obviously, a richer tagset for the MFT, where some syntactic informations are propagated from lexical items to their POS's maximal projection
- A lot of work has been done to allow for a change of the coordination annotation scheme

MFT vs FTB: Coordination

Major change between the FTB and the MFT

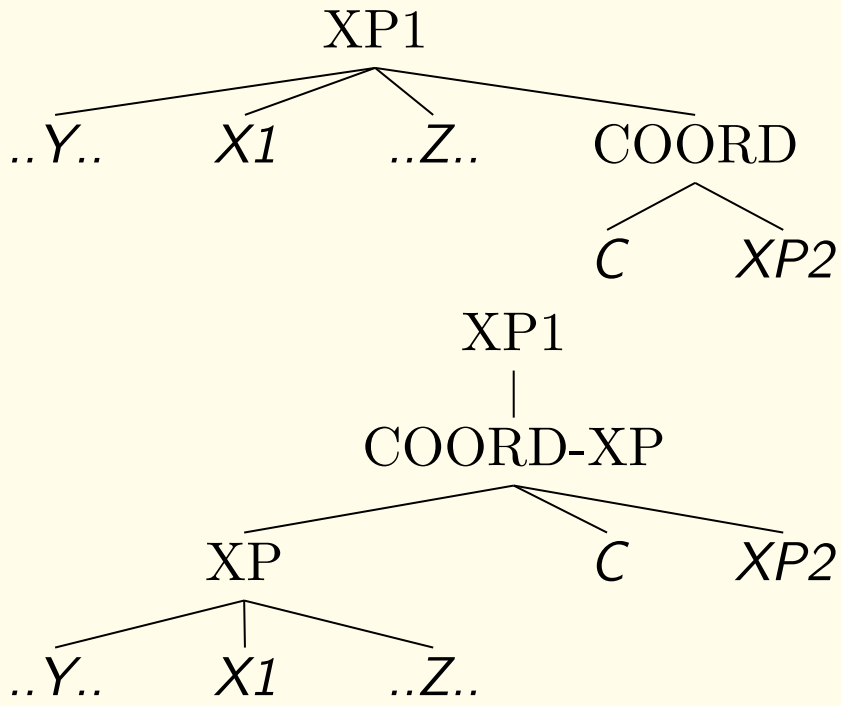


Figure: Coordinated structures in the general case, for Ftb (up) and Mft (down)

Coordination (2)

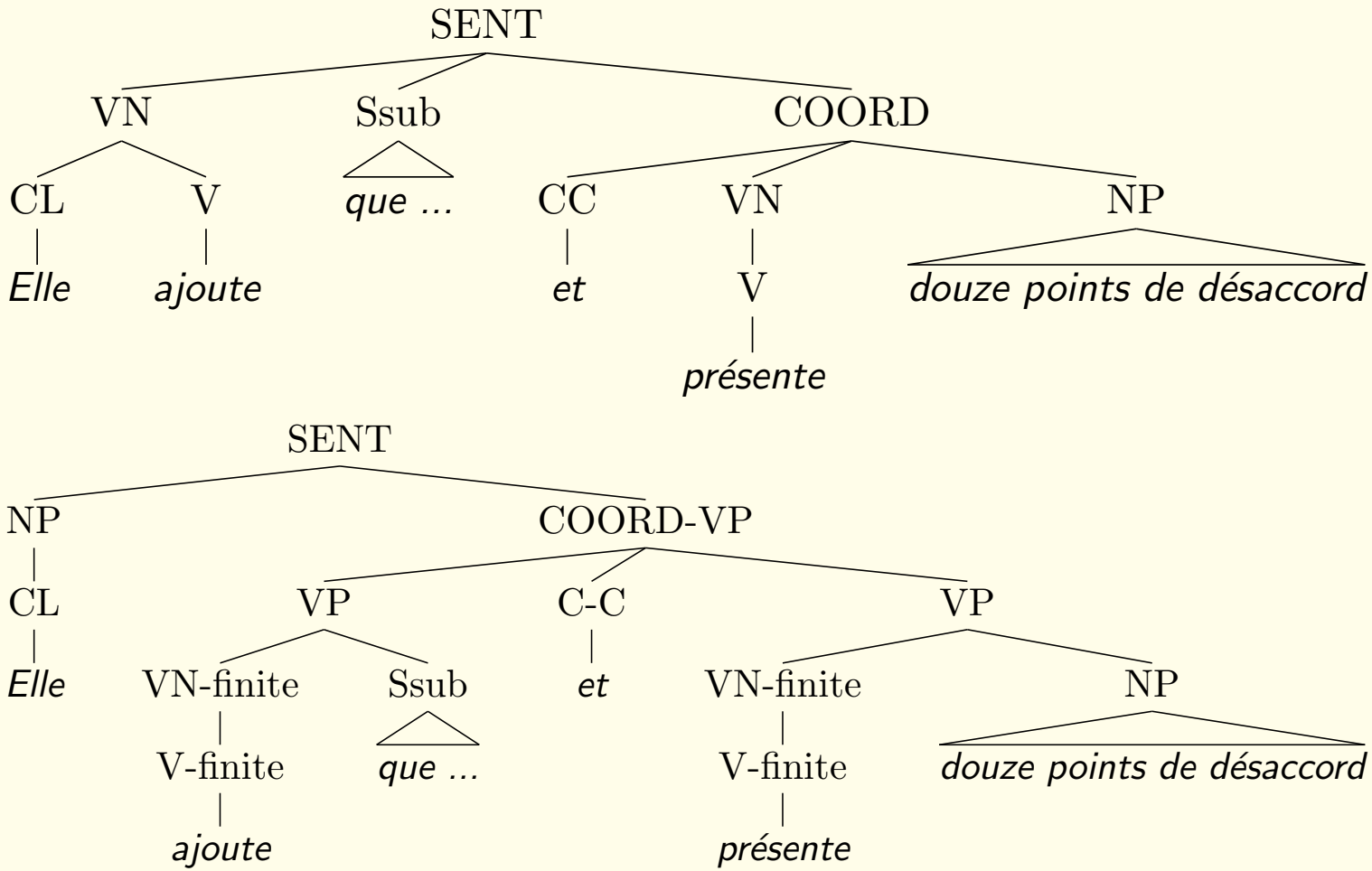


Figure: Two representations of "VP coordinations" for the sentence *She adds that ... and presents twelve sticking points*: in the Ftb (up) and in the Mft (down)

FTB vs MFT: Tagset variations

Background

- (Crabbé & Candito, 2008) showed that a specific tagset (**CC**) maximized the performance of Berkeley parser (F_1 86.42%) on the FTB

⇒ *Wh boolean propagated to A, ADV, PRO and DET*

⇒ *Mood attribute to verb tags (IMPersonal, Past Participe,...)*

⇒ *Clitics annotated with function marks (e.g. CL → CLS)*

ADJ ADJWH ADV ADVWH CC CLO CLR CLS CS DET DETWH ET I NC NPP P P+D P+PRO PONCT PREF PRO PROREL PROWH V VIMP VINF VPP VPR VS

Table: CC tagset

FTB vs MFT: Tagset variations

Background

- (Crabbé & Candito, 2008) showed that a specific tagset (**CC**) maximized the performance of Berkeley parser (F_1 86.42%) on the FTB
- The MFT has been conceived, among other things, to maximize raw parsing performance (better underlying grammar, homogeneity, etc..)

Idea : Apply MFT's choices on the FTB ?

- Unfortunately, MFT's structural transformations are not trivially applicable on the FTB (same as re-annotating the treebank)
- *However, we can extract the* **Schlu** *tagset and evaluate its impact on another treebank* **(cross parsing experiments)**
- *And, we can also extract the intersection of both treebanks and evaluate parsing results on it (Treebanks intersection experiments)*

Outline

1. Data Set : FTB and MFT
2. **Experimental Protocol**
3. Cross Parsing experiments
4. Treebanks Intersection *(ongoing work)*
5. Discussion on lexicalized models

Experimental Protocol

Objective

⇒ *Evaluate parsing on treebanks in their "optimal" mode*

⇒ *Then cross the tagsets :* **Schlu** *on the FTB and* **CC** *on the MFT*

Evaluation metrics

- *Labeled Brackets F_measure (hereafter, parseval)*
- *Unlabeled Dependencies, (Lin, 95)'s algorithm*
- *All scores are calculated on sentences of length <=40*

Experimental Protocol

Parsers

- All parsers are run without any special tweaking
- Collins' model based parsers and Berkeley include a French morphology mode for unknown words (Arun & Keller, 05)
- Head percolation table : adapted from (Dybro-Johansen, 04), argument-adjunct distinction table : based on functional labels

Training Data

- Canonical split for the MFT (430/530/3800)
- For the FTB, 1st 10% as Gold, next 10% as Dev, the rest for Training (1235/1235/9881)
- All compounds are merged (the grammar of compounds is not regular), same as all former experiments reported on FTB based data.

Outline

1. Data Set : FTB and MFT
2. Experimental Protocol
3. **Cross Parsing experiments**
4. Treebanks Intersection *(ongoing work)*
5. Discussion on lexicalized models

Back to experiment

To sum up

- We want to evaluate different parsing models

$\Rightarrow$ *using two different metrics*

$\Rightarrow$ *on two different treebanks*

- We also want to evaluate the impact of their respective tagsets

$\Rightarrow$ *So we will cross these tagsets and try to see what this teaches to us*

Baseline : minimal tagsets

		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
STIG-spinal	parseval	80.66	80.74
	Unlab. Dep	87.92	85.14
STIG	parseval	80.52	79.56
	Unlab. Dep	87,95	85.02
Bky	parseval	84,93	83.16
	Unlab. Dep	**90.06**	**87.29**

Table: F_1 results on tagset min

As the pure Stig model provides very similar results to spinal STIG (not statistically significant), we do not report its results anymore.

Cross parser tagset variation : Results

Parser	Parseval		Dependency	
	Mftcc	MftSch.	Mftcc	MftSch.
Collins (MX)	80.2	80.96	85.97	**87.98**
Collins (M2)	78.56	79.91	84.84	87.43
Collins (M1)	74	78.49	81.31	85.94
Charniak	82.5	82.66	86.45	86.94
Stig (Sp)	82.6	81.97	86.7	87.16
Bky	**83.96**	**82.86**	**87.41**	86.87

Table: Evaluation results : Mft-cc vs Mft-Schlu

Parser	Parseval		Dependency	
	Ftbcc	ftbSch.	Ftbcc	ftbSch.
Collins (MX)	82.52	82.65	88.96	89.12
Collins (M2)	80.8	79.56	87.94	87.87
Collins (M1)	79.16	78.51	86.66	86.93
Charniak	84.27	83.27	89.7	89.67
Stig (Sp)	81.73	81.54	88.85	89.02
Bky	**86.02**	**84.95**	**90.48**	**90.73**

Table: Evaluation results : Ftb-cc vs Ftb-Schlu

Tagsets have different impacts on evaluation metrics

tagset Schlu better to extract dependencies ?

- Dependencies scores of lexicalized parsers are most of the time better with the Schlu tagset than with the CC tagset
- On the Mftschlu, it is obvious that lexicalized parsers take advantage of its annotation scheme to learn dependencies more easily
- Structure modifications alone do not seem to be enough (baseline experiments) to explain that.
- Somehow lexicalized parsers have to be guided by the annotations (in the contrary, BKY has better dependency results on the MFT-min than the MFT-Schlu)

Tagsets have different impacts on evaluation metrics

tagset Schlu better to extract dependencies ?

- Dependencies scores of lexicalized parsers are most of the time better with the Schlu tagset than with the CC tagset
- On the Mftschlu, it is obvious that lexicalized parsers take advantage of its annotation scheme to learn dependencies more easily
- Structure modifications alone do not seem to be enough (baseline experiments) to explain that.
- Somehow lexicalized parsers have to be guided by the annotations (in the contrary, BKY has better dependency results on the MFT-min than the MFT-Schlu)

tagset CC better for constituency evaluation ?

- In all FTB variants, the tagset CC provides the best performance whereas the situation is not so obvious for the MFT w.r.t lexicalized parsers.

Outline

1. Data Set : FTB and MFT
2. Experimental Protocol
3. Cross Parsing experiments
4. **Treebanks Intersection (ongoing work)**
5. Discussion on lexicalized models

Treebank Intersection (ongoing work)

Both treebanks have a common base

- 3885 sentences are the same modulo very small tokenization error
  0.1% of errors (on around 100 000 tokens), mostly in one named entity (i.e. “la Cinq” vs “la_Cinq”)
- 1000 sentences from the MFT are not present in the FTB, the MFT is thus not a subset of the FTB

Idea

- Evaluate parser performance and annotation schemes on the same subset of sentences
- with the two best performing parsers (Berkeley’s and Charniak’s parser)

$\Rightarrow$ *Beware: Small data set ((T) 3112/(D) 389/(G) 384)*

Treebank Intersection

	tagset min		tagset cc		tagset Schlu	
Parser (metric)	ftbmin	Mftmin	Ftbcc	Mftcc	Ftbschlu	Mftschlu
BKY (parseval)	81.38	81.78	82.53	82.34	81.31	82.11

Table: Cross parsing evaluation on intersected treebanks

Constituency evaluation

- min : MFT raw annotation scheme leads to higher results
- cc : CC tagset leads to overall higher results especially in the FTB
- Schlu : Complete annotation MFT's scheme brings a penalty on BKY's parser (not enough data to learn a proper latent grammar ?)

Treebank Intersection

	tagset min		tagset cc		tagset Schlu	
Parser (metric)	ftbmin	Mftmin	Ftbcc	Mftcc	Ftbschlu	Mftschlu
BKY (parseval)	81.38	81.78	82.53	82.34	81.31	82.11
Char. (parseval)	80.51	80.88	81.39	81.1	81.51	81.82

Table: Cross parsing evaluation on intersected treebanks

Constituency evaluation

- min : MFT raw annotation scheme leads to higher results
- cc : CC tagset leads to overall higher results especially in the FTB

⇒ *Charniak's parser benefits from any information added to the initial minimum annotation scheme, either from the structure point of view (see mftmin vs mftschlu (+1 pt)) or from the tagset point of view (see ftbmin vs ftbschlu (+1 pt))*

Treebank Intersection

	tagset min		tagset cc		tagset Schlu	
Parser (metric)	ftbmin	Mftmin	Ftbcc	Mftcc	Ftbschlu	Mftschlu
BKY (parseval)	81.38	81.78	82.53	82.34	81.31	82.11
Char. (parseval)	80.51	80.88	81.39	81.1	81.51	81.82
BKY (Dep. based)	86.75	85.64	86.43	86.13	86.55	87.31
Char. (Dep. based)	85.79	84.11	86.48	85.27	88.18	87.6

Table: Cross parsing evaluation on intersected treebanks

Dependency based evaluation

- min & CC: The flatter annotation scheme of the FTB leads to better dependency extraction

$\Rightarrow$ *Head rule set designed for this set of experiments not optimum for more hierarchical structures with less labels (underlying grammar more ambiguous, too coarse) ?*

- Schlu : All scores improve, yet Charniak outperforms BKY's parser in both treebanks. It is unclear why Charniak has better performance on the FTBschlu.

Treebank Intersection

	tagset min		tagset cc		tagset Schlu	
Parser (metric)	ftbmin	Mftmin	Ftbcc	Mftcc	Ftbschlu	Mftschlu
BKY (parseval)	81.38	**81.78**	**82.53**	82.34	81.31	**82.11**
Char. (parseval)	80.51	80.88	81.39	81.1	81.51	81.82
BKY (Dep. based)	**86.75**	85.64	86.43	86.13	86.55	87.31
Char. (Dep. based)	85.79	84.11	**86.48**	85.27	**88.18**	87.6

Table: Cross parsing evaluation on intersected treebanks

- Point made in the crossparsing experiment is somehow confirmed : tagsets have different impacts on metrics (CC for constituency and Schlu for dependency)
- on THIS small subset, lexicalized parsers are best used with the Schlu tagset. As shown by a learning curve, BKY is likely to outperform Charniak given more data.

Outline

1. Data Set : FTB and MFT
2. Experimental Protocol
3. Cross Parsing experiments
4. Treebanks Intersection *(ongoing work)*
5. **Discussion on lexicalized models**

Discussion

- In the literature, Collins' models (via Dan Bikel's implementation) have very often been used as the main instance of lexicalized parsers (not considering STIG parsing of Chinese)
- Results were not considered as very successful ((Arun & Keller, 2005) for French, (Coraza et al, 2004) for Italian, (Dubey & Keller, 2003) for German)

$\Rightarrow$ *Treebanks flatness, small data size, free word order were used as hypothesis to explain these somehow disappointing experiments with the Collins' model 2*

Is the parsing model the problem ?

- Verify this hypothesis on previous work's French data
- Check what is the impact of the lexicon on models with the same set of heuristics (no argument-adjunct distinction table) and same parameters

Discussion : Is the parsing model the problem ?

Parser	FtbArun	MftSchlu
Arun (acl05)	**80.45**	-
Arun (this paper)	**81.08**	-
Schluter (pacling07)	-	**79.95**
Collins (Mx)	81.5	80,96
Collins (M2)	79.36	79,91
Collins (M1)	77.82	-
Charniak	82.35	82,66
Stig (Sp)	80.94	81,86
Bky	84.03	82.86

Table: Labeled bracket scores on Arun's Ftb version and on the Mft

- Our head rules set is used for "Arun (this paper)"
- Arun's bigram flat provides similar results as the model X
- Charniak's and Berkeley parser are still performing better (same order of performance as in our own treebank, Arun's is bigger and less consistant)
- Our genuine adaptation of the model 2 provides the same performance as Schluter's own implementation

Discussion : Impact of the lexicon

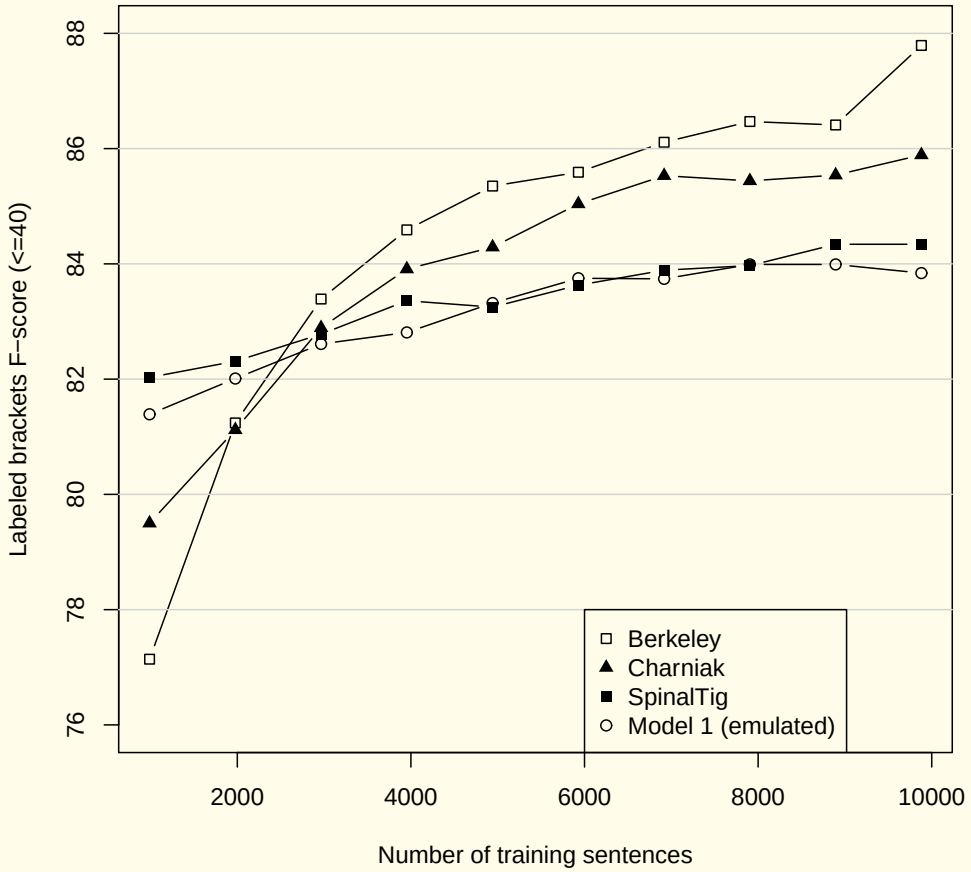


Figure: Learning Curve experiment results for parsers in perfect tagging mode without arg-adjunct distinction table

Conclusion

- We carried out a complete study on French treebanks, porting parsing models that had never been adapted to French before
- We stressed out the impact of the tagset on evaluation metrics
- By porting many lexicalized models, we showed that the argument about the usefulness of lexicalization for French could benefit from the inclusion of lexicalized models that exhibit the same order of performance as for English

Thank you and Danke !

A note on the exotic parsing models

Model X

- In Collins' generative models, the head is first generated then the Modifier Non Terminal Nodes (MNTs)
- Model 2 and Model X differ in the way they condition the generation of a MNT given a specific CONTEXT (and others features not showed here).
- MNTs p.c : $P(M_i|P, H, w_h, t_h, CONTEXT, ...)$

Model 2 context
$= map(M_i)$

Model X context
$= < M_{i-1}, ..., M_{i-k} >$

$$map(M_i) = \left\{ \begin{array}{ll} +START+ & \text{if } i = 0 \\ CC & \text{if } M_i = CC \\ +PUNC+ & \text{if } M_i =, \\ & \text{or } M_i =: \\ +OTHER+ & \text{otherwise} \end{array} \right\}$$

- This model is undocumented but present in Dan Bikel's source code.

A note on the exotic parsing models (2)

Spinal STIG

- Given a head rules set and an argument-adjuncts percolation table, a TAG can be extracted from a treebank.
- (Chiang, 00) introduced a new operation to derive flat treebank structures, the sister-adjunction. Only modifier trees can be sister-adjoined on any given node.

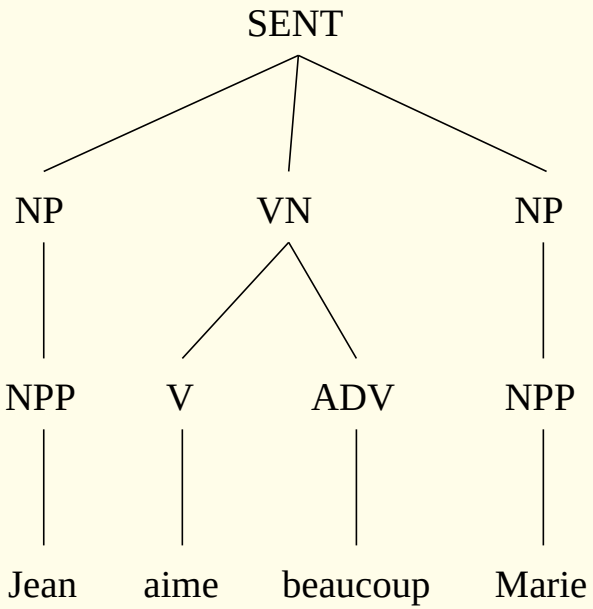
$\Rightarrow$ *Without an arg-adjunct distinction table, all trees are modifier and consist of a spine from a lexical anchor to its maximum projection*

A note on the exotic parsing models (2)

Spinal STIG

- Given a head rules set and an argument-adjuncts percolation table, a TAG can be extracted from a treebank.
- (Chiang, 00) introduced a new operation to derive flat treebank structures, the sister-adjunction. Only modifier trees can be sister-adjoined on any given node.

$\Rightarrow$ *STIG Extraction Process : (1) Parse tree*

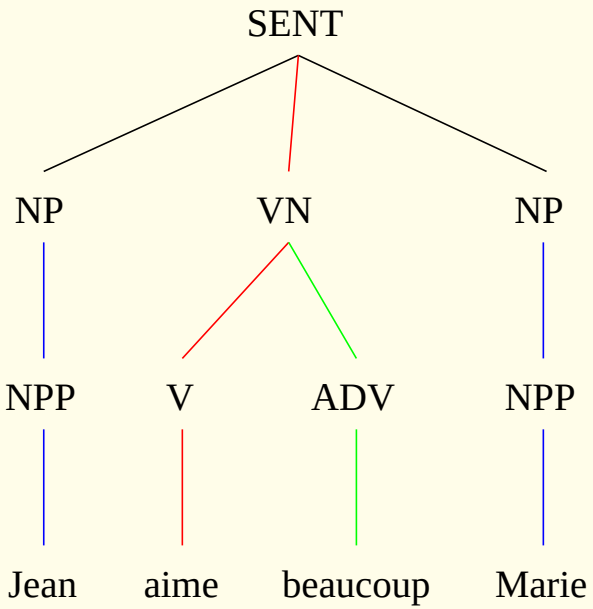


A note on the exotic parsing models (2)

Spinal STIG

- Given a head rules set and an argument-adjuncts percolation table, a TAG can be extracted from a treebank.
- (Chiang, 00) introduced a new operation to derive flat treebank structures, the sister-adjunction. Only modifier trees can be sister-adjoined on any given node.

$\Rightarrow$ *STIG Extraction Process : (2) Head Annotation*

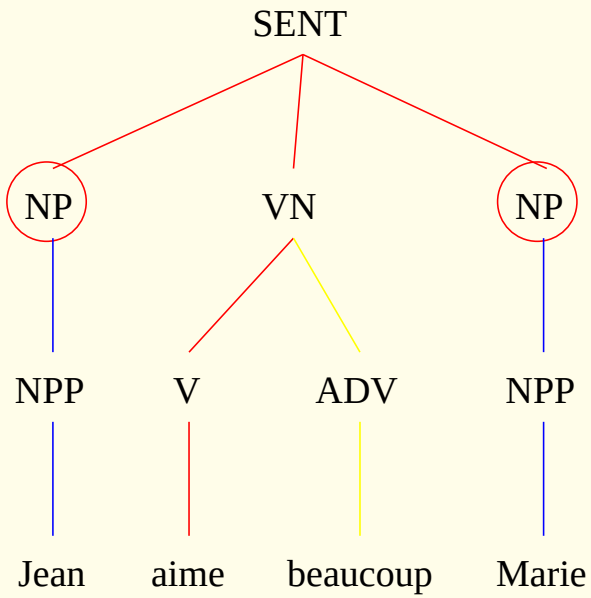


A note on the exotic parsing models (2)

Spinal STIG

- Given a head rules set and an argument-adjuncts percolation table, a TAG can be extracted from a treebank.
- (Chiang, 00) introduced a new operation to derive flat treebank structures, the sister-adjunction. Only modifier trees can be sister-adjoined on any given node.

$\Rightarrow$ *STIG Extraction Process : (3) Argument Annotation*

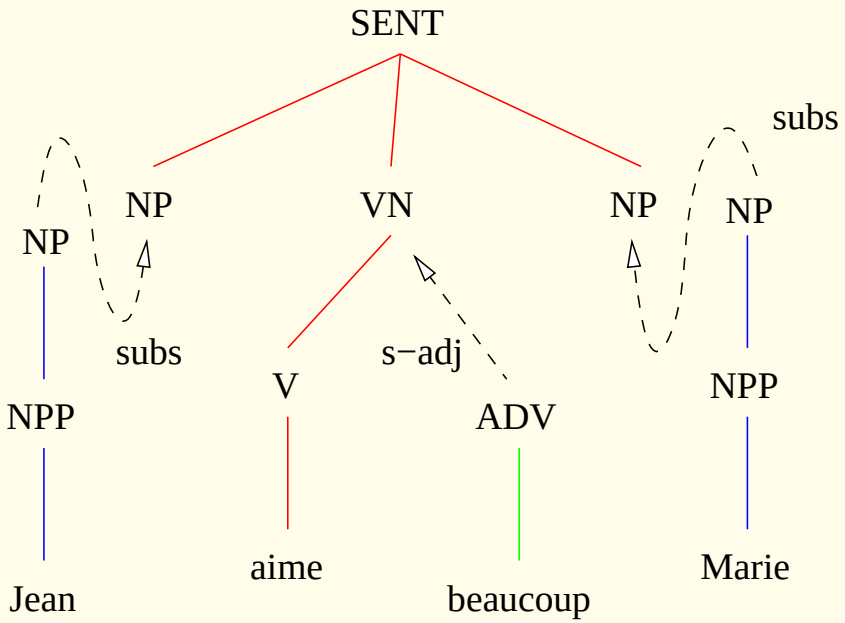


A note on the exotic parsing models (2)

Spinal STIG

- Given a head rules set and an argument-adjuncts percolation table, a TAG can be extracted from a treebank.
- (Chiang, 00) introduced a new operation to derive flat treebank structures, the sister-adjunction. Only modifier trees can be sister-adjoined on any given node.

⇒ *STIG Extraction Process : (4) Grammar Induction*

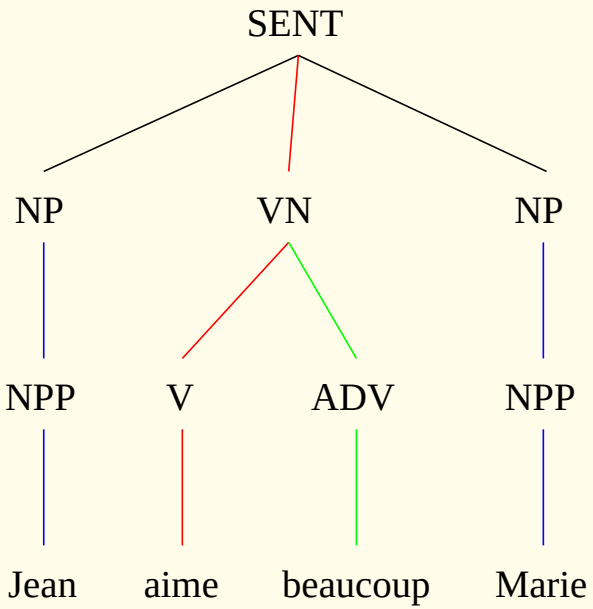


A note on the exotic parsing models (2)

Spinal STIG

- Given a head rules set and an argument-adjuncts percolation table, a TAG can be extracted from a treebank.
- (Chiang, 00) introduced a new operation to derive flat treebank structures, the sister-adjunction. Only modifier trees can be sister-adjoined on any given node.

$\Rightarrow$ *Spinal STIG Extraction Process : (2) Head Annotation*

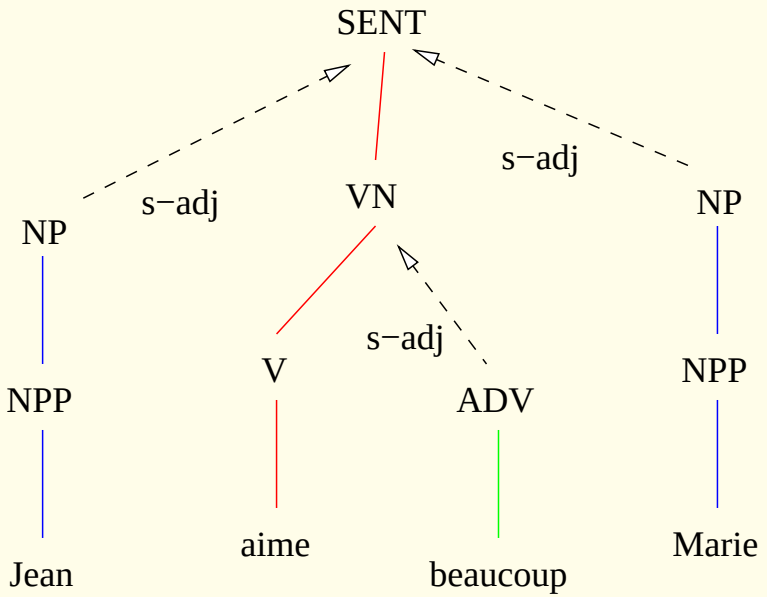


A note on the exotic parsing models (2)

Spinal STIG

- Given a head rules set and an argument-adjuncts percolation table, a TAG can be extracted from a treebank.
- (Chiang, 00) introduced a new operation to derive flat treebank structures, the sister-adjunction. Only modifier trees can be sister-adjoined on any given node.

$\Rightarrow$ *Spinal STIG Extraction Process : (3) Grammar Induction*



A note on the exotic parsing models (2)

Spinal STIG

- Given a head rules set and an argument-adjuncts percolation table, a TAG can be extracted from a treebank.
- (Chiang, 00) introduced a new operation to derive flat treebank structures, the sister-adjunction. Only modifier trees can be sister-adjoined on any given node.

$\Rightarrow$ *All extracted trees are now spinal and derived by the sister adjunction*

- Elementary trees are split between templates and lexical anchor

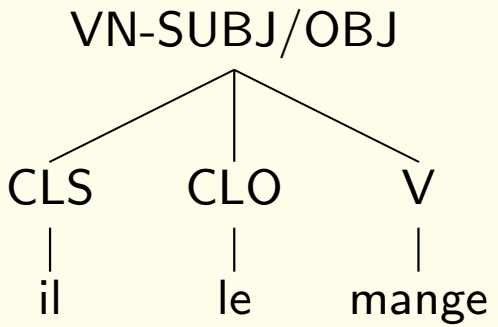
$\Rightarrow$ *Allows for very compact grammars: 83 tree templates for the FTB-CC (9881 sent.) and 424 for the TIGER German treebank (46448)*

- Not documented as well, this property of David Chiang's parser results from an unwanted side effect of his extraction algorithm

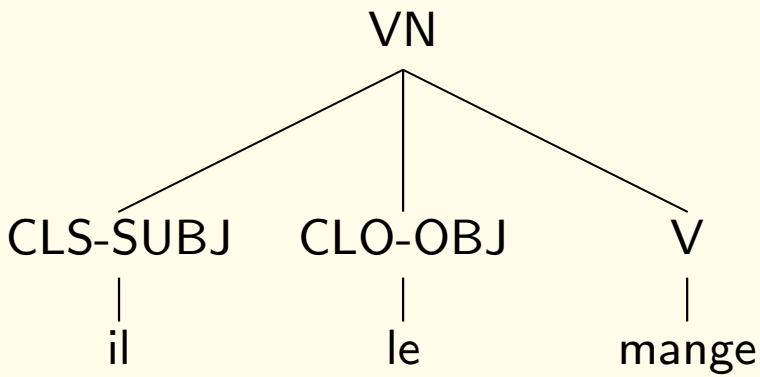
Getting our hand dirty (1)

Treebank annotation scheme : the case of clitics

- in the ftb, Clitics do not have any particular func. label (the dominating node VN handling the label
- so Benoit and Marie implemented two things
  - Clitics tag modification : (CL il) (CL le) (V mange) -> (CLS il) (CLO le) (V mange)
  - Lowering down functional labels : (CLS-Subj il) (CLO-obj le) (V mange)



regular scheme (CC)



Clitics down (CC+Clit)

Getting our hand dirty (2)

performances of Collins' model and these 2 schema

- "Weird results" :

tagset	Model 2	Model X
CC	80.8	82.52
CC+Clit	81.17	82.18

- *All results are statistically significant but with high p-value for the model X*
- Explanation ? in the case of model X in CC+clit, there're no modifier nodes before the V (CLS-A CLO-A V-Head) , so the generation of the argument does depend only on the head whereas in the CC without clit we have (CLS CLO V-head) so the model learns pretty well that a CLO is more supposed to follow an CLS