Programme 2015

Alexis Nasr

March 13, 2015

Big Picture

- Propose generic NLP tools
  - Accurate
  - Multi-lingual
  - Oral and written input

Standard pipeline achitecture

1. Automatic Speech Transcription

---

2. Sentence Boundary detection
3. Tokenization
4. Part of Speech Tagging
5. Syntactic Parsing

---

6. Coreference Resolution
7. Semantic Parsing
8. Discourse Parsing

Some problems

- Some decisions are taken too early in the pipeline
  - Postpone them
- Treebanks are too small for modeling some phenomena
  - Use external resources

Small Picture

Four problems in relation with the syntactic parser:

1. Tokenization of Grammatical Complex Words
2. Syntactic Lexicon
3. Selectional Preferences
4. Sentence Boundaries Detection

Tokenization of Grammatical Complex Words

- The decision to group a sequence of tokens as a single lexical unit is often taken very early in the NLP pipeline
- The choice can be difficult to make and should be done by the parser:
  - *Je mange bien que je n'aie plus faim*
  - *Je pense bien que je n'ai plus faim*
- Introduce a morphological dependecy to represent this kind of structure: bien $\overset{MORPH}{\longleftarrow}$ que
- Such a dependency is built by the parser

Preliminary Results

The 8 most frequent ADV-que structures and their ambiguity

ADV-que	complex conj.	other
alors que	88	12
autant que	86	14
bien que	40	60
depuis que	98	2
encore que	20	80
maintenant que	51	49
plus que	29	71
tant que	20	80
total	432	368

Preliminary Results

ADV-que	recall	prec.	f-meas.
alors que	0.95	0.97	0.96
bien que	0.86	0.75	0.80
encore que	0.72	0.80	0.76
maintenant que	0.81	1.00	0.90
total	0.87	0.92	0.90

Some problems

- exogenous v/s endogenous compounds
  - endogenous compound : the PoS of the compound corresponds to the PoS of one element (ex : [bien/ADV que/CSU]/CSU)
  - exogenous compounds : none of the elements has the PoS of the compound (ex : [en/PRE fait/NOM]/ADV)
- In some cases, the decision is taken by the tagger
  en/PRE fait/NOM il/CLI en/PRO fait/VRB trop/ADV

Introduction of a syntactic lexicon in the parser

- Some parsing decisions depend on the syntactic properties of the lexical entries
- in the sentences :
  - *Je mange bien que je n'aie plus faim*
  - *Je pense bien que je n'ai plus faim*
- syntactic properties of *penser* and *manger* are important to predict the correct parse
- treebanks are not large enough to learn subcat frames

Introduction of a syntactic lexicon in the parser

- but, we have syntactic lexica that contain this information
- however, the domain of locality of subcat frames exceed the size of the configurations that the parser sees.
- parse recombining using ILP
- quite successful (80.84 $\rightarrow$ 85.26 SFAS).
- but, the method is complex and time consuming

Introduction of a syntactic lexicon in the parser

- Define new lexico-syntaxic features (LSF): `OBJ, AOBJ, DEOBJ, QOBJ ...`
- Derive a syntactic lexicon from existing ones: `LEMMA LSF* (donner OBJ AOBJ)`
- Define new first order feature template: `LSF -fct-> POS (OBJ -obj-> N)`

Selectional Preferences

- Some parsing decisions depend on the semantic (lexical) nature of the words
- in the sentences :
  - *Il mange une* **escalope** *à la* **crème**
  - *Il* **mange** *une escalope à la* **cantine**
- lexical affinities of (VàN, mange, cantine) and (NàN, escalope, crème) are important to make the right choice
- treebanks are not large enough to learn such lexial affinities

Use Raw Corpora

- Parse Raw Corpus
- Compute lexical affinities
- Inject in the parser :
  - parse recombining using ILP
  - quite successful (87.81 $\rightarrow$ 92.32 SCAS).
  - but, the method is complex and time consuming

Selectional Preferences

- Introduce selectional preferences through features in the parser
  - First experiments were not successful
  - Not enough new features to modify the output of the parser ?
- Use word embeddings to model lexical affinities ?

Sentence boundaries detection

- Vicious circle:
  - the parser needs to know sentence boundaries
  - sentence boundary detector needs syntax
- Challenging problem: the parser cannot run on very long sequences.
- Two steps approach:
  - segment the speech transcription into large segments which boundaries can be reliably predicted
  - parse the segments to detect syntactic boundaries