LAP: The CLARINO Language Analysis Portal

Emanuele Lapponi, Stephan Oepen, Arne Skjærholt, and Erik Velldal

University of Oslo,
Department of Informatics

October 16, 2015

In this talk:

- Introduction & high level goals
- Design & Implementation: Galaxy
- LAF as a model for tool interchange in LAP
- Tool integration & versioning with the LAP tree
- Reaching out to other research communities
- Current state of development and future plans

Introduction

LAP:

- A portal providing easy access to NLP tools
- Unlike other processing environments:
  - LAP is not web-service based
  - Tools run on a high-capacity compute cluster
  - Annotation representation and interchange format

Introduction

LAP:

- A portal providing easy access to NLP tools
- Unlike other processing environments:
  - LAP is not web-service based
  - Tools run on a high-capacity compute cluster
  - Annotation representation and interchange format
- Part of an ongoing PhD project that investigates how NLP can benefit SSH research

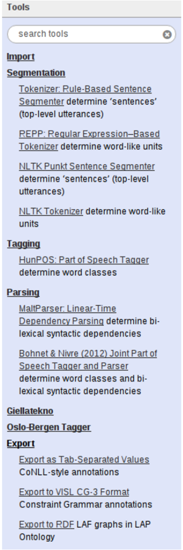
Galaxy

A web-app platform for accessing and configuring tools, organizing datasets and annotations, and share results.

Galaxy

Galaxy

A web-app platform for accessing and configuring tools, organizing datasets and annotations, and share results.



Tools

search tools

Import

Segmentation

- Tokenizer: Rule-Based Sentence Segmenter determine ‘sentences’ (top-level utterances)
- REPP: Regular Expression–Based Tokenizer determine word-like units
- NLTK Punkt Sentence Segmenter determine ‘sentences’ (top-level utterances)
- NLTK Tokenizer determine word-like units

Tagging

- HunPOS: Part of Speech Tagger determine word classes

Parsing

- MaltParser: Linear-Time Dependency Parsing determine bi-lexical syntactic dependencies
- Bohnet & Nivre (2012) Joint Part of Speech Tagger and Parser determine word classes and bi-lexical syntactic dependencies

Giellatekno

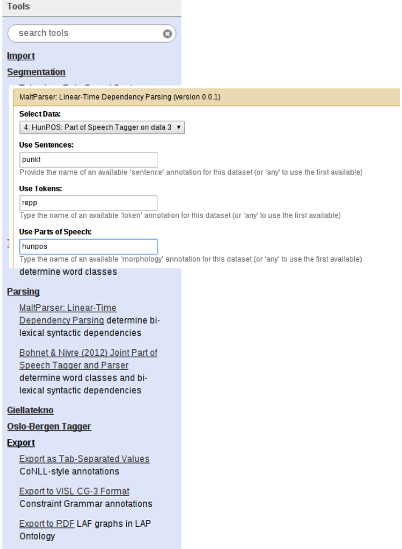
Oslo-Bergen Tagger

Export

- Export as Tab-Separated Values CoNLL-style annotations
- Export to VISL CG-3 Format Constraint Grammar annotations
- Export to RDF LAF graphs in LAP Ontology

Galaxy

A web-app platform for accessing and configuring tools, organizing datasets and annotations, and share results.



Tools

search tools

Import

Segmentation

MaltParser: Linear-Time Dependency Parsing (version 0.0.1)

Select Data: 4: HunPOS: Part of Speech Tagger on data 3

Use Sentences: punkt

Provide the name of an available 'sentence' annotation for this dataset (or 'any' to use the first available)

Use Tokens: repp

Type the name of an available 'token' annotation for this dataset (or 'any' to use the first available)

Use Parts of Speech: hunpos

Type the name of an available 'morphology' annotation for this dataset (or 'any' to use the first available)

determine word classes

Parsing

- MaltParser: Linear-Time Dependency Parsing determine bi-lexical syntactic dependencies
- Bohnet & Nivre (2012) Joint Part of Speech Tagger and Parser determine word classes and bi-lexical syntactic dependencies

Giellatekno

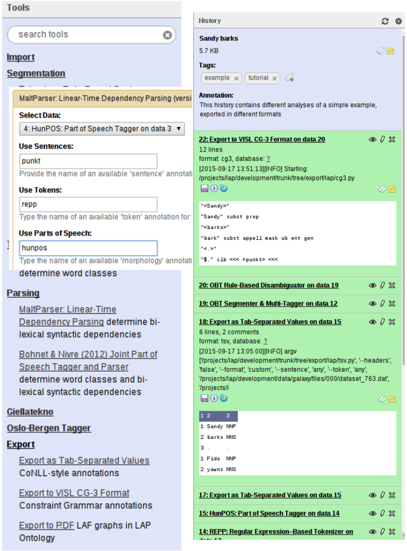
Oslo-Bergen Tagger

Export

- Export as Tab-Separated Values CoNLL-style annotations
- Export to VISL CG-3 Format Constraint Grammar annotations
- Export to RDF LAF graphs in LAP Ontology

Galaxy

A web-app platform for accessing and configuring tools, organizing datasets and annotations, and share results.



Tools

search tools

Import

Segmentation

MaltParser: Linear-Time Dependency Parsing (versi

Select Data: 4: HunPOS: Part of Speech Tagger on data 3

Use Sentences: punkt

Provide the name of an available 'sentence' annotati

Use Tokens: repp

Type the name of an available 'token' annotation for

Use Parts of Speech: hunpos

Type the name of an available 'morphology' annotati

determine word classes

Parsing

- MaltParser: Linear-Time Dependency Parsing determine bi-lexical syntactic dependencies
- Bohnet & Nivre (2012) Joint Part of Speech Tagger and Parser determine word classes and bi-lexical syntactic dependencies

Giellatekno

Oslo-Bergen Tagger

Export

- Export as Tab-Separated Values CoNLL-style annotations
- Export to VISL CG-3 Format Constraint Grammar annotations
- Export to RDF LAF graphs in LAP Ontology

History

Sandy barks

5.7 KB

Tags: example × tutorial ×

Annotation:
This history contains different analyses of a simple example, exported in different formats

22: Export to VISL CG-3 Format on data 20

12 lines
format: cg3, database: ?
[2015-09-17 13:51:13][INFO] Starting: /projects/lap/development/trunk/tree/export/lap/cg3.py

```
"<Sandy>"
"Sandy" subst prop
"<barks>"
"bark" subst appell mask ub ent gen
"<.>"
"$." clb <<< <punkt> <<<
```

20: OBT Rule-Based Disambiguator on data 19

19: OBT Segmenter & Multi-Tagger on data 12

18: Export as Tab-Separated Values on data 15

6 lines, 2 comments
format: tsv, database: ?
[2015-09-17 13:05:00][INFO] argv ['/projects/lap/development/trunk/tree/export/lap/tsv.py', '--headers', 'false', '--format', 'custom', '--sentence', 'any', '--token', 'any', '/projects/lap/development/data/galaxy/files/000/dataset_763.dat', '/projects/l

```
1 2     3
1 Sandy NNP
2 barks NNS
3 .     .
1 Fido  NNP
2 yawns NNS
```

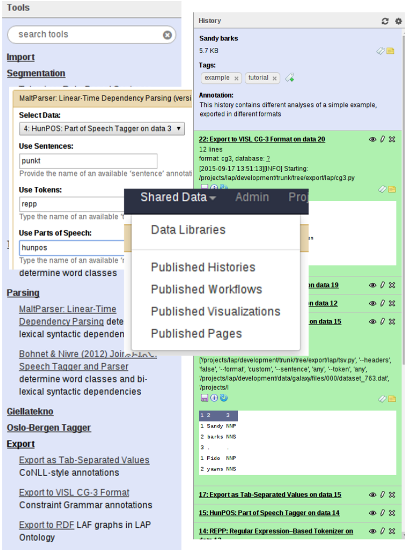
17: Export as Tab-Separated Values on data 15

15: HunPOS: Part of Speech Tagger on data 14

14: REPP: Regular Expression–Based Tokenizer on

Galaxy

A web-app platform for accessing and configuring tools, organizing datasets and annotations, and share results.



Tools

search tools

Import

Segmentation

MaltParser: Linear-Time Dependency Parsing (versi

Select Data:

4: HunPOS: Part of Speech Tagger on data 3 ▾

Use Sentences:

punkt

Provide the name of an available 'sentence' annotati

Use Tokens:

repp

Type the name of an available 't

Use Parts of Speech:

hunpos

Type the name of an available 'r

determine word classes

Parsing

MaltParser: Linear-Time Dependency Parsing dete lexical syntactic dependen

Bohnet & Nivre (2012) Joint Speech Tagger and Parser determine word classes and bi-lexical syntactic dependencies

Giellatekno

Oslo-Bergen Tagger

Export

Export as Tab-Separated Values CoNLL-style annotations

Export to VISL CG-3 Format Constraint Grammar annotations

Export to RDF LAF graphs in LAP Ontology

History

Sandy barks

5.7 KB

Tags:

example × tutorial ×

Annotation:

This history contains different analyses of a simple example, exported in different formats

22: Export to VISL CG-3 Format on data 20

12 lines

format: cg3, database: ?

[2015-09-17 13:51:13][INFO] Starting: /projects/lap/development/trunk/tree/export/lap/cg3.py

on data 19

on data 12

on data 15

['/projects/lap/development/trunk/tree/export/lap/tsv.py', '--headers', 'false', '--format', 'custom', '--sentence', 'any', '--token', 'any', '/projects/lap/development/data/galaxy/files/000/dataset_763.dat', '/projects/l

```
1 2     3
1 Sandy NNP
2 barks NNS
3 .     .
1 Fido  NNP
2 yawns NNS
```

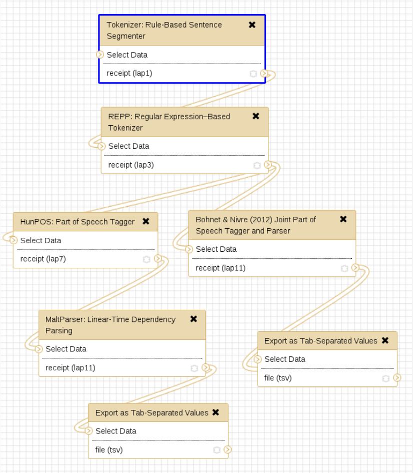
17: Export as Tab-Separated Values on data 15

15: HunPOS: Part of Speech Tagger on data 14

14: REPP: Regular Expression–Based Tokenizer on

Galaxy

A GUI to build (potentially complex) workflows



Tokenizer: Rule-Based Sentence Segmenter ✖
Select Data
receipt (lap1)

REPP: Regular Expression–Based Tokenizer ✖
Select Data
receipt (lap3)

HunPOS: Part of Speech Tagger ✖
Select Data
receipt (lap7)

Bohnet & Nivre (2012) Joint Part of Speech Tagger and Parser ✖
Select Data
receipt (lap11)

MaltParser: Linear-Time Dependency Parsing ✖
Select Data
receipt (lap11)

Export as Tab-Separated Values ✖
Select Data
file (tsv)

Export as Tab-Separated Values ✖
Select Data
file (tsv)

Tool interchange format

Requirements:

- Stand-off
- Scalable in terms of coverage of linguistic information and data volume
- Granular: on-demand access of relevant annotations

Tool interchange format

Data model: **L**inguistic **A**nnotation **F**ramework [2]

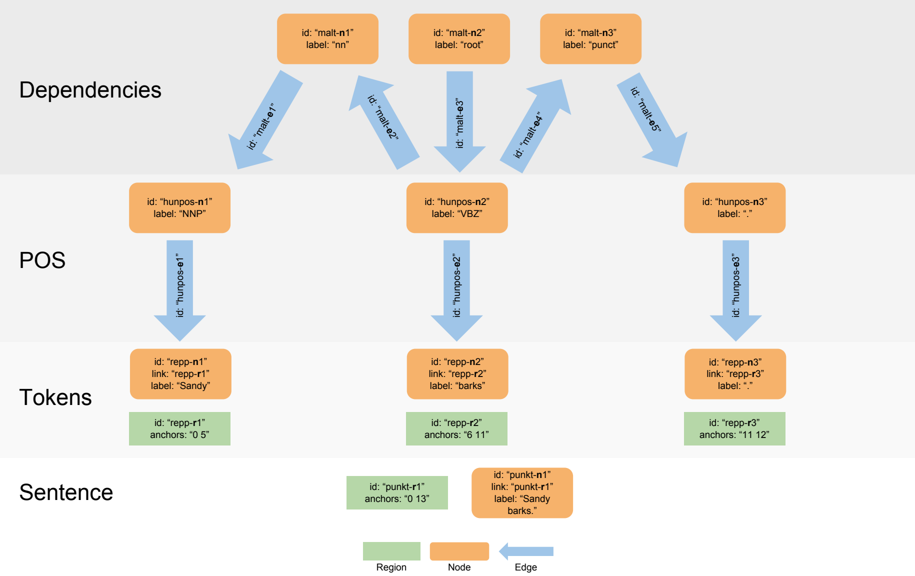
- Stand-off: text regions linked to a graph that describes them
- Content agnostic
- Flexible structure

Tool interchange format

Implementation: MongoDB

- Records describing the structural LAF elements: Regions, Nodes and Edges
- Flexible data-access

Tool interchange format



Dependencies

id: “malt-n1”
label: “nn”

id: “malt-n2”
label: “root”

id: “malt-n3”
label: “punct”

id: “malt-e1”

id: “malt-e2”

id: “malt-e3”

id: “malt-e4”

id: “malt-e5”

POS

id: “hunpos-n1”
label: “NNP”

id: “hunpos-n2”
label: “VBZ”

id: “hunpos-n3”
label: “.”

id: “hunpos-e1”

id: “hunpos-e2”

id: “hunpos-e3”

Tokens

id: “repp-n1”
link: “repp-r1”
label: “Sandy”

id: “repp-n2”
link: “repp-r2”
label: “barks”

id: “repp-n3”
link: “repp-r3”
label: “.”

id: “repp-r1”
anchors: “0 5”

id: “repp-r2”
anchors: “6 11”

id: “repp-r3”
anchors: “11 12”

Sentence

id: “punkt-r1”
anchors: “0 13”

id: “punkt-n1”
link: “punkt-r1”
label: “Sandy barks.”

Region

Node

Edge

Tool interchange format

Not in competition with richer end-user formats!

Tool integration and versioning: the LAP tree

A LAP tool is made of:

- binaries for the actual annotator (i.e. the B&N parser)
- a *wrapper* that communicates with MongoDB

Tool integration and versioning: the LAP tree

A LAP tool is made of:

- binaries for the actual annotator (i.e. the B&N parser)
- a *wrapper* that communicates with MongoDB

Which means:

- Different programming languages
- Different virtual machines and interpreters
- Different versions

Tool integration and versioning: the LAP tree

The LAP Tree

- A version controlled repository of the core LAP parts (i.e. those that transcend Galaxy and the OS)
- Easily relocatable
- Enables reproducibility of experiments performed with historical versions of tools

Reaching out to other research communities

If we build it, will they come? [3]

Reaching out to other research communities

Our position:

- Start out with actual research questions
- Work jointly with SSH researchers
- Investigate how and to what degree this work can be generalized into workflows

Reaching out to other research communities

Our position:

- Start out with actual research questions
- Work jointly with SSH researchers
- Investigate how and to what degree this work can be generalized into workflows

So far:

- Joint work with Political Scientists
- Data-driven analysis of plenary debate speeches in the European Parliament [1]

Reaching out to other research communities

Talk of Europe

- A project that aims at curating EP datasets to linked data

Our contribution:

- State-of-the-art syntacto-semantic annotations in rdf triples
  (and possible ontological means to connect them to the ToE graph)

Current state of development and future plans

LAP, currently:

- A feide- and eduGAIN-accessible development instance
- HPC-ready tools for English, Sami and Norwegian
- Tabulated, cg3 and rdf export
- Basic user documentation

Current state of development and future plans

Short- to mid-term goals:

- Broaden the range of processing types (e.g. deep semantic parsing)
- Preprocessing tools for e.g. xml-datasets
- Export interfaces with other CLARINO platforms such as Corpuscle and Glossa

Thank you!

Thank you!

`https://lap.hpc.uio.no/`

References

B. Høyland, J.-F. Godbout, E. Lapponi, and E. Velldal.
Predicting party affiliations from European Parliament debates.
In *Proceedings of the 52nd Meeting of the Association for Computational Linguistics: Workshop on Language Technologies and Computational Social Science*, page 56 – 60, Baltimore, MD, USA, 2014.

N. Ide and K. Suderman.
The Linguistic Annotation Framework: A standard for annotation interchange and merging.
Language Resources and Evaluation, (forthcoming), 2013.

J. v. Zundert.
If you build it, will we come? Large scale digital infrastructures as a dead end for digital humanities.
Historical Social Research, 37(3), 2012.