UDMorph: Morphosyntactically Tagged UD Corpora

Maarten Janssen

ÚFAL, MFF, Charles University

Universal Dependencies

- One of the most successful linguistic resources initiatives
  - 10 years, 500 collaborator, 200 treebank, 100 languages
- Various strong points
  - Same tags for all languages
  - Fully accessible resources that can be easily used in NLP task
  - Based on established standards
  - Periodic releases with rigorous verification checks
  - Integrated in various highly popular tool, incl. UDPIPE and spaCy
- Success still largely accidental
  - Right place at the right time
  - Various attempts to recreate the success in related fields have failed

Treebanks and Tagged Corpora

- UD is a framework for treebanks
  - Only valid treebanks are accepted into the infrastructure
- UD (also) provides standardization for morphosyntax
  - (originally Google) Universal POS tags
- Used outside of UD for POS tagged corpora
  - Masakhane: African grassroots organisation
  - RESTAURE project for Alsatian dialects
- Not acceptable as UD resources
  - Yet very useful to have
- UDMorph
  - Provide an infrastructure for tagged corpora following UD guidelines

Motivation

- Support for TEITOK
  - Language independent environment for tokenized TEI corpora
  - Currently using UDPIPE and NeoTag for tagging
  - Ideally support for many languages
- Massively multilingual corpora
  - Developed at UFAL
  - Tatoeba - sentences in 500 languages
  - Only a fraction has NLP support

UDMorph

- Follows the structure of UD
  - CoNLL-U format repositories with UPOS, Feats, Lemma, Misc
  - Languages codes following ISO-639 with variant naming where needed
  - Periodic releases of static resources
  - Verification with UD script that do not rely on dependencies
  - No attempts to improve or change UD
- Provide POS taggers
  - REST service following the specifications of UDPIPE
- TEITOK interface
  - Searchable version of repositories
  - Possible tool to help people create new UDMorph corpora

TEITOK Corpora

- Tokenized annotated TEI files
  - CoNNL-U attributes over token nodes
  - Fully interchangeable with CoNLL-U
  - Mostly hidden in UDMorph
- Searchable version of the repositories
  - Can contain much more metadata
- Intermediate format for conversion of corpora
  - Many existing scripts to convert and correct
  - Currently 54 corpora in TEITOK
- Guided creation of new UDMorph corpora
  - GUI for correcting automatic annotations
  - Bootstrapping: manual tag, train, automatic tag, correct, train, repeat

REST POS Taggers

- GUI Interface
  - Paste and tag, similar to UDPIPE
  - Sends request to REST service and show result
  - Pulls from a list of services with URL, format, language, family, and features
  - REST service list can be used directly in pipelines
- Homogenized taggers
  - Various types of taggers all producing UDPIPE-style output
  - All UDPIPE languages
  - Taggers locally trained on the TEITOK corpora (as CoNLL-U)
  - Locally run available taggers, potentially with on-the-fly conversion to UD
  - Currently 150 languages

CoNLL-U Repositories

- Parallel set-up to UD
  - Git repositories with CoNLL-U files
  - https://github.com/UDMorph/
- Some maintained as repositories
  - Clones of repositories hosted elsewhere
  - Repositories directly edited at UDMorph
- Some exported from TEITOK corpora
  - Tagged resources converted by us
  - Corpora created in TEITOK
- Made available only with consent
  - Currently still only 4 repositories
  - Currently not cloning UD treebanks, since they can be used in parallel

Periodic releases

- Stable versions of repositories
  - Released on LINDAT
  - Potentially on HuggingFace as well
- Only valid repositories
  - All UD scripts that do not rely on dependencies
  - Adapted scripts that remove the checks on dependencies
  - Additional UDMorph specific scripts
  - Non-validating repositories can stay in the Git
- Only starting up
  - No releases yet

Transitional stage

- Currently most resources assembled by us
  - Often not validating
- Imperfect taggers and data
  - Not all resources and tagger provide features or lemmas
  - Sometimes impossible to fully convert to UD: no distinction between SCONJ / CCONJ
  - REST list indicates the status of each of the fields provided by resource/tagger
  - Imperfect data should over time get replaced by valid alternatives
- Bootstrapped improvement
  - Tag new resource with existing imperfect tagger
  - Correct and complete results
  - Distribute fully compatible data

Depends on community

- Resources have to be made by community
  - We can help to convert, create, or integrate
  - Good way to make new resources available and visible
  - Long-term storage of tagger data
  - Hopefully NLP community will add better taggers trained on data
- Please contribute!
  - Open source resources with UD tagging
  - Available resources in other format with conversion table
  - Existing REST taggers (in UD or with conversion table)
  - Installable taggers (in UD or with conversion table)
  - Ask for a TEITOK corpus to build new resources
  - https://github.com/UDMorph/udmorph_contributions