NorNE – Norwegian Named Entities

The resource

Named entity annotations on top of Norwegian Dependency Treebank. Created as a collaboration between Schibsted Media Group, Språkbanken at the National Library of Norway and the Language Technology Group at the University of Oslo.

The NorNE corpus is published under the same license as the Norwegian Dependency Treebank.

About the Norwegian Dependency Treebank (NDT)

The Norwegian Dependency Treebank (NDT) consists of text which is manually annotated with morphological features, syntactic functions and hierarchical structure. The formalism used for the syntactic annotation is dependency grammar. With a few exceptions, the syntactic analysis follows *Norsk referensegrammatikk* "Norwegian Reference Grammar".[1]

NDT consists of two parts, one in Norwegian Bokmål (nob) and one in Norwegian Nynorsk (nno). Both parts contain around 300.000 tokens, and contain a mix of different non-fictional genres.

See the NDT webpage for more details.

About the Named Entity annotations

NDT has been extended with NER annotations. The texts, tokenization and syntactic annotations from the original NDT has not been changed in any way.

The annotated files are distributed in two different collections:

1. ndt/, the same files as in the NDT resource, extended with entity annotations.
2. ud/, the files in ndt/ in a train/dev/test split, as distributed in the Universal Dependencies project.

Extended with entity annotations. More details on the splits can be found in the documentation of the Norwegian Bokmål UD project. Each subdirectory contains a folder for the two variants of Norwegian, Bokmål (nob) and Nynorsk (nno), respectively.

Entity types

The following types of entities are annotated:

- Person (PER)
- Organisation (ORG)
- Location (LOC)
- Geo-political entity (GPE)
- Product (PROD)
- Event (EVT)
- Miscellaneous (MISC)
- Derived (DRV)

Furthermore, all GPE entities are additionally sub-categorized as being either ORG or LOC, with the two annotation levels separated by an underscore:

[1] Jan Terje Faarlund, Svein Lie and Kjell Ivar Vannebo, *Norsk referansegrammatikk*, Universitetsforlaget, Oslo, 1997.

- GPE_LOC: Geo-political entity, with a locative sense ("John lives in *Spain*")
- GPE_ORG: Geo-political entity, with an organisation sense ("*Spain* declined to meet with Belgium")

The two special types GPE_LOC and GPE_ORG can easily be altered depending on the task, choosing either the more general GPE tag or the more specific LOC/ORG tags, conflating them with the other annotations of the same type. This means that the following sets of entity types can be derived:

- 7 types, deleting _GPE: **ORG**, **LOC**, PER, PROD, EVT, DRV, MISC
- 8 types, deleting LOC_ and ORG_: **ORG**, **LOC**, **GPE**, PER, PROD, EVT, DRV, MISC
- 9 types, keeping all types: **ORG**, **LOC**, **GPE_LOC**, **GPE_ORG**, PER, PROD, EVT, DRV, MISC

The class distribution is as follows, broken down across the data splits of the UD version of NDT, and sorted by total counts (i.e. the number of examples, not tokens within the spans of the annotatons):

Type	Train	Dev	Test	Total
PER	4033	607	560	5200
ORG	2828	400	283	3511
GPE_LOC	2132	258	257	2647
PROD	671	162	71	904
LOC	613	109	103	825
GPE_ORG	388	55	50	493
DRV	519	77	48	644
EVT	131	9	5	145
MISC	8	0	0	0

Entity definitions

- **Person:** Real or fictional characters and animals.
- **Organization:** Any collection of people, such as firms, institutions, organizations, music groups, sports teams, unions, political parties etc.
- **Location:** Geographical places, buildings and facilities.
- **Geo-political entity:** Geographical regions defined by political and/or social groups. A GPE entity subsumes and does not distinguish between a nation, its region, its government, or its people.
- **Product:** Artificially produced entities are regarded products. This may include more abstract entities, such as speeches, radio shows, programming languages, contracts, laws and ideas.
- **Event:** Festivals, cultural events, sports events, weather phenomena, wars, etc. Events are bounded in time and space.
- **Derived:** Words (and phrases?) that are derived from a name, but are not a name in themselves. They typically contain a full name and are capitalized, but are not proper nouns. Examples (fictive) are "Brann-treneren" ("the Brann coach") or "Oslo-mannen" ("the man from Oslo").
- **Miscellaneous:** Names that do not belong in the other categories. Examples are animals species and names of medical conditions. Entities that are manufactured or produced are of type Products, whereas thing naturally or spontaneously occurring are of type Miscellaneous.

Annotation principles

1. A *name* in this context is close to Saul Kripke's definition of a name, in that a name has a unique reference and its meaning is constant (there are exceptions in the annotations, e.g. "Regjeringen" ("Government")).
2. It is the usage of a name that determines the entity type, not the default/literal sense of the name.
3. If there is an ambiguity in the type/sense of a name, then the the default/literal sense of the name is chosen (following Markert and Nissim, 2002).

Annotation scheme

The entities are annotated using the IOB2 format:

- **Beginning:** The first token of an entity is annotated B-<TYPE>, e.g. B-PER for the first token in a person name
- **Inside:** The following tokens of an entity are annotated I-<TYPE>, e.g. I-PER for the second token in a person name
- **Outside:** Tokens outside an entity are annotated O

Example:

```
1  John       ...   name=B-PER
2  Towner     ...   name=I-PER
3  Williams   ...   name=I-PER
4  is         ...   name=O
...
```

File format

The texts are on the CONLL-U format, a tab-separated columnar format, as described below. Named entity annotations are found in the MISC field (10th column), on the format name=<TYPE>.

From Universal dependency's description of the CONLL-U format:

Annotations are encoded in plain text files (UTF-8, using only the LF character as line break, including an LF character at the end of file) with three types of lines:

1. Word lines containing the annotation of a word/token in 10 fields separated by single tab characters; see below.
2. Blank lines marking sentence boundaries.
3. Comment lines starting with hash (#).

Sentences consist of one or more word lines, and word lines contain the following fields:

1. ID: Word index, integer starting at 1 for each new sentence; may be a range for multiword tokens; may be a decimal number for empty nodes.
2. FORM: Word form or punctuation symbol.
3. LEMMA: Lemma or stem of word form.
4. UPOS: Universal part-of-speech tag.
5. XPOS: Language-specific part-of-speech tag; underscore if not available.
6. FEATS: List of morphological features from the universal feature inventory or from a defined language-specific extension; underscore if not available.
7. HEAD: Head of the current word, which is either a value of ID or zero (0).
8. DEPREL: Universal dependency relation to the HEAD (root iff HEAD = 0) or a defined language-specific subtype of one.

9. DEPS: Enhanced dependency graph in the form of a list of head-deprel pairs.
10. MISC: Any other annotation.

The fields DEPS and MISC replace the obsolete fields PHEAD and PDEPREL of the CoNLL-X format. In addition, we have modified the usage of the ID, FORM, LEMMA, XPOS, FEATS and HEAD fields as explained below.

The fields must additionally meet the following constraints:

- Fields must not be empty.
- Fields other than FORM and LEMMA must not contain space characters.
- Underscore (_) is used to denote unspecified values in all fields except ID. Note that no format-level distinction is made for the rare cases where the FORM or LEMMA is the literal underscore – processing in such cases is application-dependent. Further, in UD treebanks the UPOS, HEAD, and DEPREL columns are not allowed to be left unspecified.