

# Determining Named Entity Class Senses in WordNet and GermaNet

## Goal

Add lexical information to Named-Entity classes in *WordNet* and *GermaNet*:

**'name-bearing' or not:** the fact that a word can refer to an object, person, etc. that can be named with an individual name

**'role' or 'type':** being a 'type' is to have a property persistently over time and being a 'role' is to have a property over a finite period of time

## Data and Pre-processing

- 450 noun lemmas from *WordNet*
  - 300 classified as 'name-bearing'
  - 150 classified as 'not name-bearing'
- Data annotation by *CrowdFlower*
- Libraries used for editing *WordNet*:
  - *JAWS* (Java API for WordNet Search)
  - *extJWNL* (extended Java WordNet Library)



Leo Born, Catarina Cramer, Julian Gerhard  
Department of Computational Linguistics  
Heidelberg University, Germany

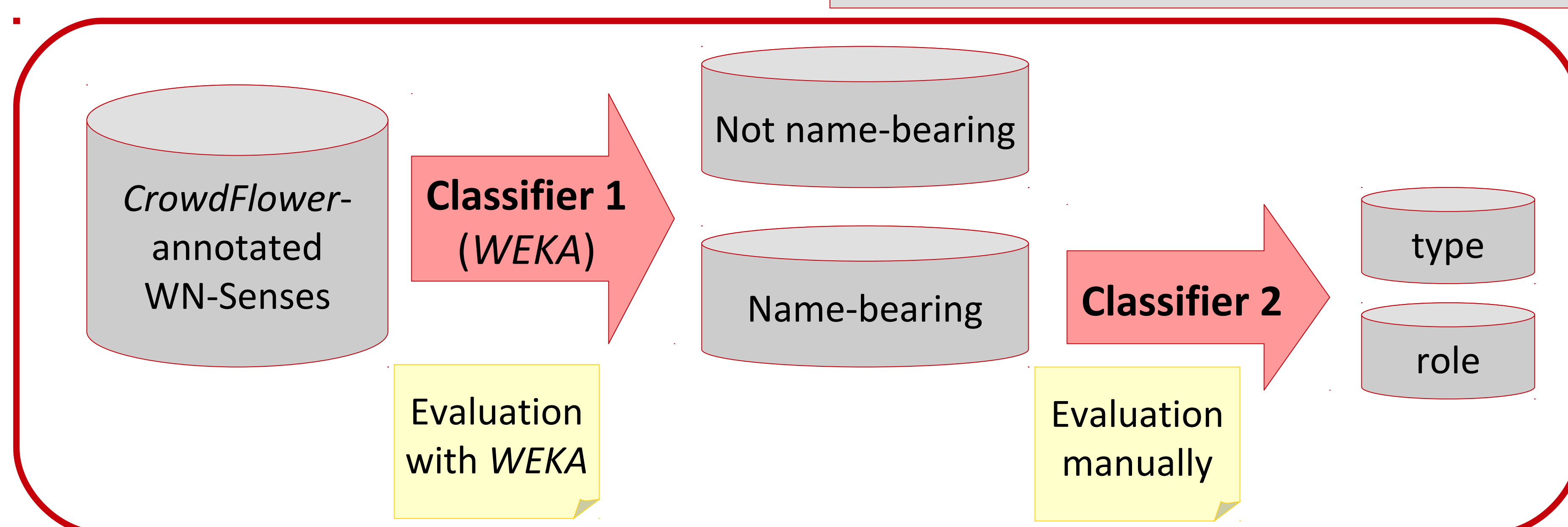
## Classifier 2

**Idea:** Fetch the sentences out of *UKWaC* which contain our name-bearing nouns and use corpus features for classification.

**Approach:** Run the classifier in two classification steps and evaluate manually.

### Features:

- Several regular expressions from *Rudify* like "is/was no longer a/the" which are indicators for 'role'
- A few ontological features from *WordNet* hierarchy:
  - Some lemmas are typically role, like 'worker'. If the current noun contains 'worker' it will be classified as 'role'
  - After the first step, all words are classified as 'type' or 'role'. However, if in the hypernym structure of a target word x another target word y is found that is classified as 'role', x will be classified as 'role' afterwards, too.



## Classifier 1

**Idea:** Derive information from the gloss of the given noun sense and the *WordNet* hierarchy of hyponyms and hypernyms.

**Approach:** Automatic classification and evaluation with *WEKA*.

### Features:

- Gloss content, e.g. 'someone' or 'somebody' are strong indicators for a name-bearing noun
- Ontological features from *WordNet* hierarchy, for example:
  - Number of hyponyms
  - Number and percentage of hyponyms with instances etc.

## Noun

- **S: (n) usher, guide** (someone employed to conduct others)
- **S: (n) guide** (someone who shows the way by leading or advising)
  - **direct hyponym / full hyponym**
    - **S: (n) cicerone** (a guide who conducts and informs sightseers)
    - **S: (n) tour guide** (a guide who leads others on a tour)
  - **direct hypernym / inherited hypernym / sister term**
    - **S: (n) leader** (a person who rules or guides or inspires others)
  - **derivationally related form**

Exemplary representation of the word 'guide' in *WordNet*

## Experimental Setup

|                     | Training | Dev. Test | Test | Sum  |           |
|---------------------|----------|-----------|------|------|-----------|
| <b>Classifier 1</b> | 551      | 220       | 219  | 990  | instances |
| <b>Classifier 2</b> | 800      | -         | 378  | 1178 | instances |

Note that we collected all *WordNet* synonyms of each sense from *UKWaC*, that's why there are more instances in Classifier 2.

## Evaluation

### Classifier 1:

- development test set: average precision 75%
- test set: average precision 79%  
F1-Measure 79%
- Kappa: 0.57
- Best Feature: Lexical file name

### Classifier 2 (test set):

- 'role' precision 84%, recall 58%
- 'type' precision 64%, recall 42%
- overall F1-Measure 60%
- Best feature: Regular expressions from *Rudify*

| Is...  | Classified as... |        |
|--------|------------------|--------|
|        | nb               | Not nb |
| nb     | 76               | 26     |
| Not nb | 20               | 97     |

Classifier 1: Confusion Matrix for test set

| Is... | Classified as... |      |
|-------|------------------|------|
|       | type             | role |
| type  | 49               | 13   |
| role  | 27               | 68   |

Classifier 2: Confusion Matrix for test set

## Conclusion

There are several things to do in future. One of them was that we weren't able to apply our data on *WordNet* and *GermaNet* until we could provide some good results. Nevertheless, we prepared all noun synsets from *WordNet* as a *WEKA*-classifiable .arff file and we also wrote the code for editing the dictionary and tried it out locally.

## References

- Espresso: Leveraging Generic Patterns for Automatically Harvesting Semantic Relations  
Patrick Pantel, Marco Pennacchiotti 2006
- Cross-Lingual Evaluation of Ontologies with *Rudify*  
Amanda Hicks, Axel Herold 2011
- Automatically Harvesting and Ontologizing Semantic Relations  
Patrick Pantel, Marco Pennacchiotti 2008