ANEA: Automated (Named) Entity Annotation for German Domain-Specific Texts

Anastasia Zhukova, Felix Hamborg, Bela Gipp

2nd Workshop on Extraction and Evaluation of Knowledge Entities from Scientific Documents (EEKE2021)
30 September 2021

DATA & KNOWLEDGE ENGINEERING

BERGISCHE UNIVERSITÄT WUPPERTAL

Introduction

DATA & KNOWLEDGE ENGINEERING

BERGISCHE UNIVERSITÄT WUPPERTAL

Introduction

- Named entity recognition (NER) is a well-known NLP task.
- NER datasets contain general categories, e.g., person, location, time, etc.

Problems

1. General NER reflects no categories of the other domains, e.g., technology, production
2. A small number of NLP datasets for German, i.e., a low-resources language
3. Domain NER requires annotating a dataset for training a NER model

   → a very time-consuming task

Goal

- Minimize the time of creating a domain dataset for NER in German by automating the annotation process

Research question

How to use knowledge graphs (e.g., Wiktionary) to automatically

1. extract domain terms (nouns),
2. derive entity categories,
3. annotate these terms into categories?

Domain graph

DATA & KNOWLEDGE ENGINEERING

BERGISCHE UNIVERSITÄT WUPPERTAL

Preprocessing and Wiktionary pages

German compound nouns

Sechszylindermotor (six-cylinder motor)
= sechs + Zylinder + Motor

Extracted noun phrases (NPs)

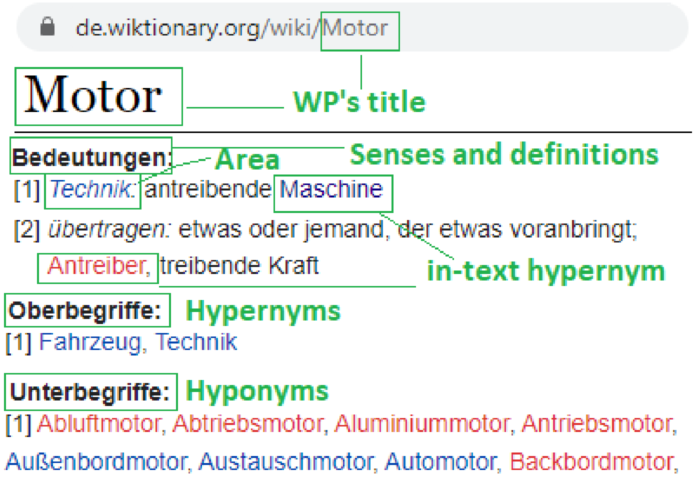
NP has a matching Wiktionary page (WP)

NP doesn't match any Wiktionary page

Extract head of NP + match the head to Wiktionary page

Properties from WPs: (1) Hypernyms, (2) Hyponyms, (3) Definitions and areas

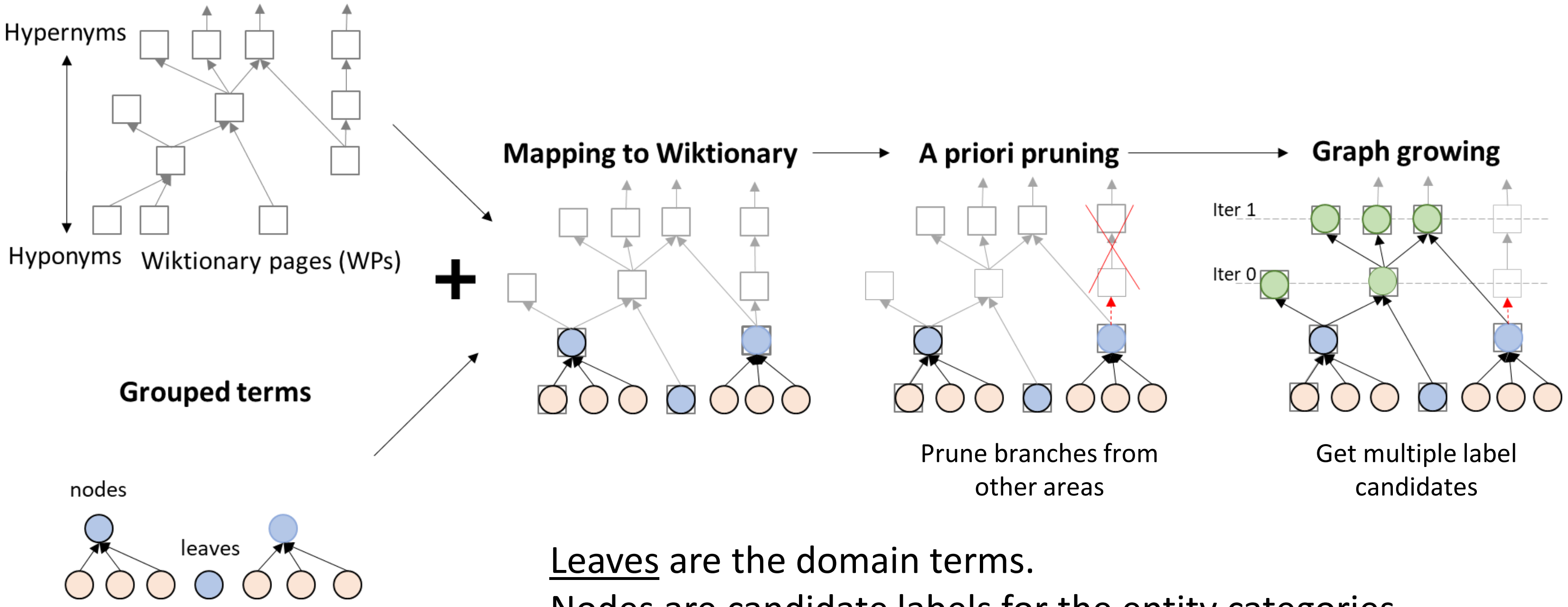
A Wiktionary page (WP) matches "Motor":



→ Use the NPs that were mapped to Wiktionary pages

Domain graph

<u>Domain graph</u> is a *Wiktionary subgraph* with nodes related to the current domain



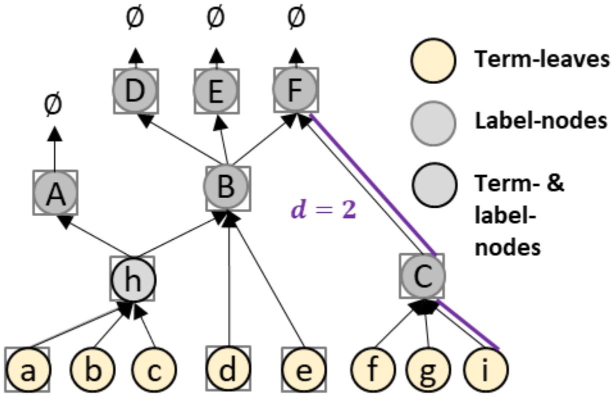
<u>Leaves</u> are the domain terms.
<u>Nodes</u> are candidate labels for the entity categories.

ANEA

DATA & KNOWLEDGE ENGINEERING

BERGISCHE UNIVERSITÄT WUPPERTAL

Candidate Entity Categories



Candidate labels per term

a → [h, A, B, D, E, F]
b → [h, A, B, D, E, F]
c → [h, A, B, D, E, F]
h → [A, B, D, E, F]
d → [B, D, E, F]
e → [B, D, E, F]
f → [C, F]; g → [C, F]
i → [C, F]

→ "transpose"

Initial (entity) categories

h: [a, b, c]
A: [a, b, c, h]
B: [a, b, c, d, e, h]
D: [a, b, c, d, e, h]
E: [a, b, c, d, e, h]
F: [a, b, c, d, e, h, f, g, i]
C: [f, g, i]

Specific	Optimal	General
(Actor, Politician)	(Person)	(Humanity)

Word embeddings:
fastText represents well out-of-vocabulary terms and labels

The quality metric:

$$\boldsymbol{Q_i = T_i \cdot L_i \cdot O_i \cdot \max(\log_2|EC_i|, 1) \cdot d_{avg_i}}$$

T_i is a mean cross-term cosine similarity
L_i is a mean label-terms cosine similarity
$O_i = T_i + L_i$ is an overall similarity
$|EC_i|$ is a number of terms in an entity category
d_{avg_i} is an average of non-zero distances between terms and a label

Optimization steps

1. Candidate filtering

1) a mean cross-term similarity too small ($T_i < 0.2$)
2) a mean label-terms similarity too small ($L_i < 0.3$)
3) EC is too broad (contains > 15% of all terms-to- annotate)
4) EC is too narrow (contains < 5 terms)

2. Resolution of full overlaps

When containing same terms, keep an EC_i with the largest Q_i

h: [a, b, c]
A: [a, b, c, h]
B: [a, b, c, d, e, h]
D: [a, b, c, d, e, h]
E: [a, b, c, d, e, h]
F: [a, b, c, d, e, h, f, g, i]
C: [f, g, i]

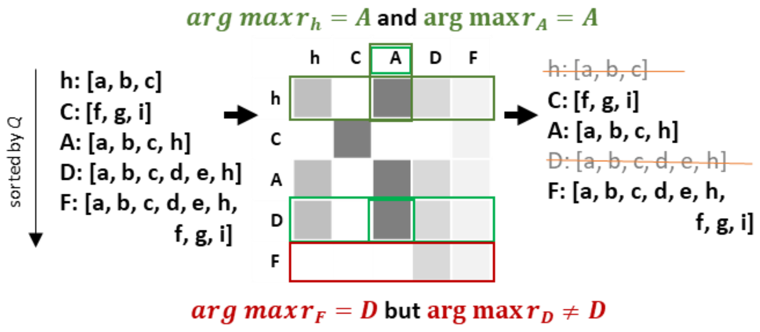
→

sorted by Q

h: [a, b, c]
A: [a, b, c, h]
D: [a, b, c, d, e, h]
~~E: [a, b, c, d, e, h]~~
~~B: [a, b, c, d, e, h]~~
F: [a, b, c, d, e, h, f, g, i]
C: [f, g, i]

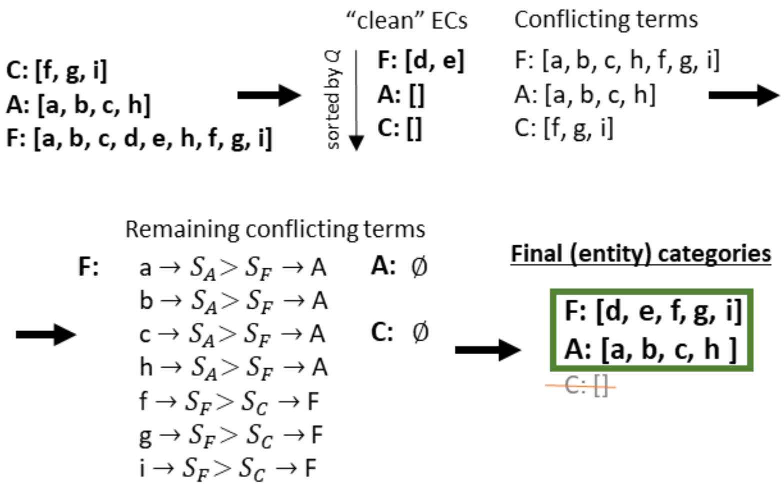
3. Resolution of the substantial overlaps

When terms overlap ≥ 50%, keep a big EC_i that is a best replacement to a small EC_j and to itself



4. Resolution of the conflicting terms

Resolve the conflicting terms to “clean” candidate EC_i with the highest overall similarity O_i



Evaluation

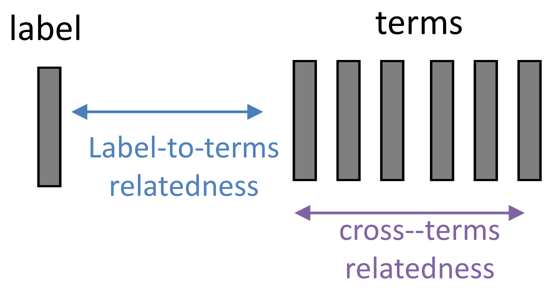
DATA & KNOWLEDGE ENGINEERING

BERGISCHE UNIVERSITÄT WUPPERTAL

Evaluation: User study

No German domain-specific dataset available →
performed a user study to evaluate the results

- *4 datasets*: processing industry, software development, databases, and travelling
- *9 native German study participants*: 4 f, 5 m, aged between 23-60
- *2-4 evaluators* per dataset
- *4 various configurations* per dataset: different number of terms-to-annotate
- *2 methods*: ANEA and a hierarchical clustering baseline

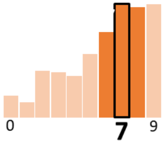
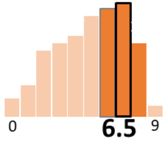
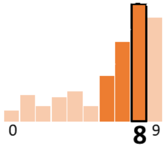
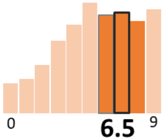
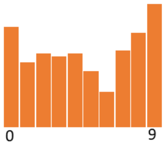
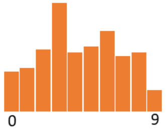
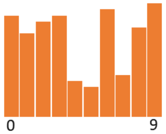
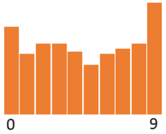


Tasks

1) Evaluate *cross-term relatedness* within a category:
   0-9 where 9 is the best
2) Evaluate *relatedness of a label to terms* in a category:
   0-9 where 9 is the best

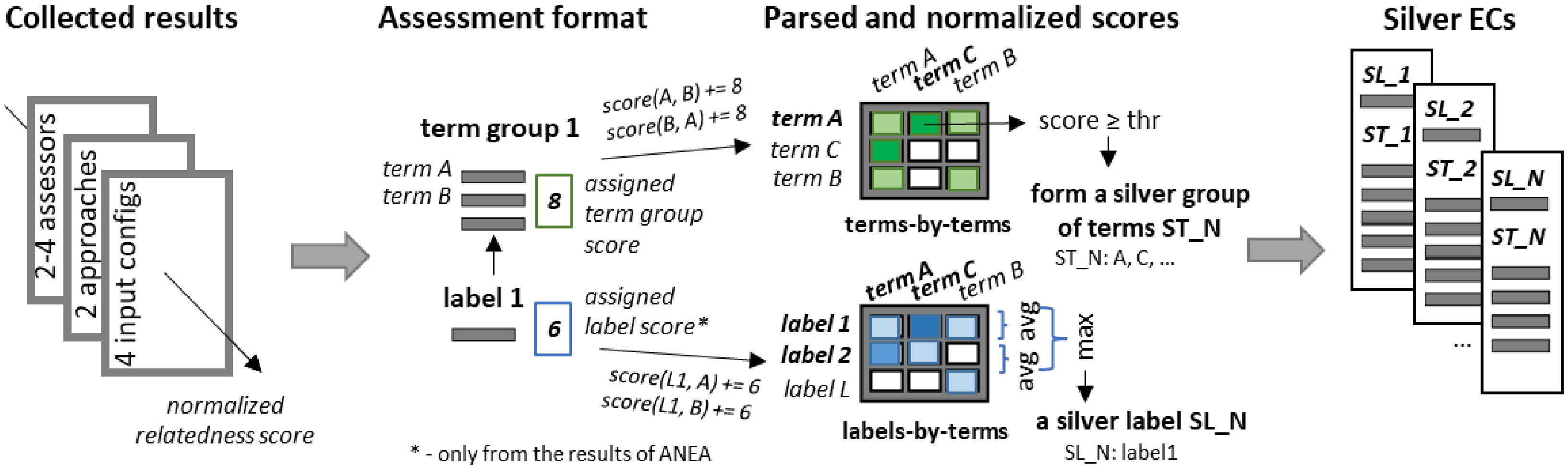
Dataset properties

- *Distribution* of the relatedness scores between the datasets *differ*.
- The most frequent score per dataset is used as *thresholds* for *creating silver datasets*.

Dataset	**Databases**	**Software development**	**Traveling**	**Processing**
All words	8161	8581	6293	7984
Terms	1209	1041	1040	552
Heads	713	673	801	328
Assessors	3	3	2	4
Cross-term relatedness				
Label-terms relatedness				

Collection of a silver dataset

Silver datasets are required to compare configurations of ANEA against it.



Evaluation methods

- Silver dataset
- Hierarchical clustering
  - A baseline for terms relatedness
- ANEA
- ANEA voting
  - A final result is derived in an ensemble/voting strategy of multiple ANEA configurations

Results

Topic	Method	# terms-to-annotate	# entity categories (ECs)	# annotated terms	ECs' average size	Term similarity	Label similarity	Average similarity
Databases	silver	420	5	113	23	7.2	7	**7.2**
	HC	253	8	52	7	7.2	--	7.2*
	ANEA	253	18	179	10	5.7	5	5.4
	ANEA voting	253-316	12	122	10	6.3	5.9	<u>***6.1***</u>
Software dev.	silver	356	6	57	10	6.2	6	**6.1**
	HC	303	15	152	10	5.5	--	5.5*
	ANEA	191	10	119	12	5	5.3	5.2
	ANEA voting	191-255	4	44	11	5.6	6.5	<u>***6.0***</u>
Traveling	silver	363	6	115	19	7.8	6.7	**7.3**
	HC	363	19	156	8	7.3	--	7.3*
	ANEA	363	22	239	11	5.4	4.8	5.1
	ANEA voting	258-363	12	146	12	6.2	5.6	<u>***5.9***</u>
Processing	silver	282	7	102	15	6.6	6.2	**6.4**
	HC	183	7	56	8	6.1	--	6.1*
	ANEA	227	16	172	11	5.3	4.9	5.1
	ANEA voting	181-282	9	157	17	5.7	5.6	<u>***5.6***</u>

ANEA voting shows <u>improvement of 13-15%</u> to the original ANEA average similarity scores.

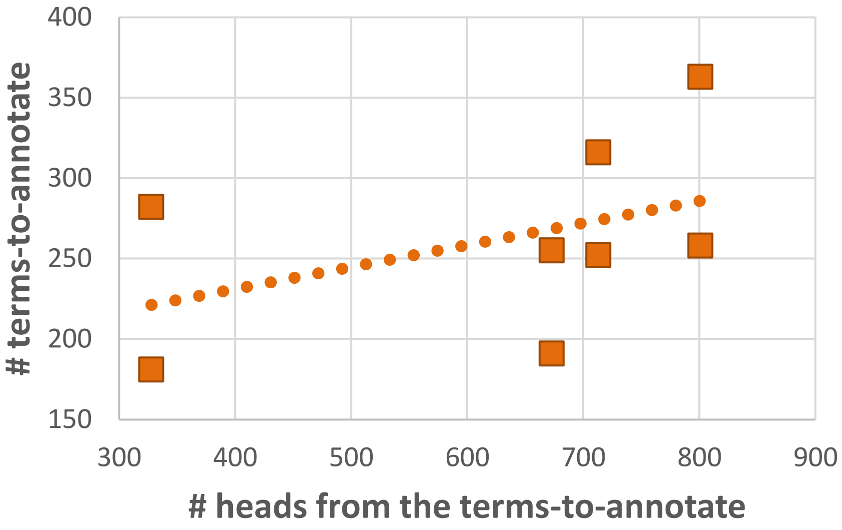
ANEA summary

- ANEA hasn't achieved the relatedness scores of the silver datasets yet.
- The voting strategy shows a significant improvement to the ANEA results

Recommended configurations for ANEA with voting:

1) $y = 158 + 0.167x$
2) $y + 50$
3) $y - 50$

where x is a number of unique heads among the terms to annotate and y is a number of terms-to-annotate by ANEA



Conclusion

Conclusion

- Proposed ANEA, i.e., an **unsupervised approach** for automated creation of a small dataset for **domain-specific NER.**
- Evaluated ANEA with a user study on **four domain datasets**.
- The produced entity categories **required less than one hour**, which is significantly **faster than manual annotation.**
- The produced entity categories are slightly worse than the silver datasets but **a voting strategy improves the scores by 13-15%.**

A suggested use case with using ANEA:

(1) annotate a small dataset,

(2) validate and improve the dataset with manual inspection,

(3) use the produced dataset in a semi-supervised or transfer learning

Questions

Anastasia Zhukova

zhukova@uni-wuppertal.de