Published as:

Smirnova, N., & Mayr, P. (2022). Evaluation of Embedding Models for Automatic Extraction and Classification of Acknowledged Entities in Scientific Documents. Proceedings of EEKE 2022. http://arxiv.org/abs/2206.10939

KB Project Report:

Mining Acknowledgement Texts in Web of Science (MinAck)

30.11.2021

Nina Smirnova, Philipp Mayr (GESIS)

Summary

The focus of our project MinAck is the detection and quantitative analysis of acknowledged entities using the FLAIR NLP-framework. We trained and implemented a named entity recognition (NER) task in a larger corpus of Web of Science (WoS) articles, which include acknowledgements. Several corpora were created: two annotated training corpora of different sizes and one acknowledgment corpus (approx. 200,000 entries), which contains acknowledgement texts for the analysis. Flair has three default training algorithms for NER, which were used for primary training: NER Model with Flair Embeddings (later on ***Flair Embeddings***) (Akbik et al., 2018), NER Model with Transformers (later on ***Transformers***) (Schweter & Akbik, 2020), Zero-shot NER with TARS (later on ***TARS***) (Halder et al., 2020). Flair Embeddings showed the best accuracy, therefore the analysis of the acknowledgement corpus was performed using a NER tagger trained with the Flair Embeddings. Our NER tagger can be tested via an online demo[1]. Analysis of the automatically extracted entities revealed differences and distinct patterns in the distribution of acknowledged entities of different types between different scientific domains. The data of the analysis is available[2].

1. Introduction

Acknowledgements in scientific papers are short texts where the author(s) "*identify those who made special intellectual or technical contribution to a study that are not sufficient to qualify them for authorship*" (Kassirer & Angell, 1991, p. 1511). The focus of our project MinAck[3] is

[1] https://mybinder.org/v2/gh/kalawinka/minack/main?labpath=example_model.ipynb

[2] All results and the description of each file: https://github.com/kalawinka/minack/tree/results

[3] https://kalawinka.github.io/minack/

the detection and quantitative analysis of acknowledged entities, i.e., the named entity recognition (NER) task in a larger corpus of Web of Science (WoS) articles, which include acknowledgements. An acknowledged entity is an object in the acknowledgment which can consist of e.g. names and surnames of individuals (also abbreviations), names of institutions and organisations, numbers, or identifiers of grants.

The analysis of acknowledgments is particularly interesting as acknowledgments may give an insight on such aspects of the scientific community as reward systems, collaboration structures, and hidden research trends (Giles & Councill, 2004). In addition, acknowledgements can help the reader to better understand the set-up and framing of a given scientific text. From the linguistic point of view, acknowledgements are unstructured text data, which automatic analysis poses interesting research and methodological problems like data cleaning, tokenization, word embedding.

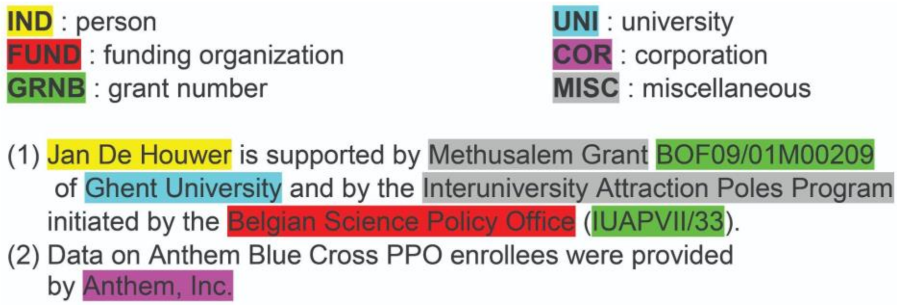
WoS is a website, which provides subscription-based access to publisher-independent global citation databases (Web of Science Group, 2021). WoS contains publications from different scientific fields. From 2008, WoS started indexing funding information (funding agencies and grant numbers) to its databases (Clarivate, 2021) (Figure 2).

The present project aims to create a method for automatic extraction and classification of acknowledged entities from acknowledgment texts and examine the correlation between the acknowledged entity category and scientific domain.

2. Approach

2.1. Methodology and framework

Two of the aims of the present project are to extract acknowledged entities from the acknowledgments corpus and ascribe them to different categories. The choice of categories was inspired by Giles and Councill (Giles & Councill, 2004, p. 17601) classification: funding agencies (FUND), corporations (COR), universities (UNI), individuals (IND). For the present project, this classification was enhanced with the MISC (miscellaneous) and grant numbers (GRNB) categories. The GRNB category was adopted from WoS funding information indexing. In the miscellaneous category fall entities, which could provide useful information, but can not be ascribed to other categories, e.g. names of the ships, names of projects, names of conferences. Figure 1 demonstrates the example of acknowledged entities of different types.



IND : person
FUND : funding organization
GRNB : grant number
UNI : university
COR : corporation
MISC : miscellaneous

(1) Jan De Houwer is supported by Methusalem Grant BOF09/01M00209 of Ghent University and by the Interuniversity Attraction Poles Program initiated by the Belgian Science Policy Office (IUAPVII/33).
(2) Data on Anthem Blue Cross PPO enrollees were provided by Anthem, Inc.

Figure 1: example of acknowledged entities. Each entity type is marked with a distinct color.

A large-scale analysis of acknowledgment texts from WoS was conducted using the FLAIR NLP Framework (Akbik et al., 2019). FLAIR is an open-sourced NLP framework and built on PyTorch (Paszke et al., 2019). "*The core idea of the framework is to present a simple, unified interface for conceptually very different types of word and document embeddings*" (Akbik et al., 2019, p. 54). FLAIR has shown better accuracy for NER tasks using pretrained datasets in comparison with other open source NLP tools (Akbik, n.d., 2021).

FLAIR provides the possibility to create a custom NER model (Akbik, 2021b; Chauhan, 2020). Creating a custom NER tagger allows us to accomplish acknowledged entity recognition and acknowledged entity classification in one step. As a result, the model should have been able to recognize six entity types: funding agencies (FUND), corporations (COR), universities (UNI), individuals (IND), grant numbers (GRNB) and miscellaneous (MISC).

Flair has 3 default training algorithms for NER, which were used for primary training: NER Model with Flair Embeddings (later on ***Flair Embeddings***) (Akbik et al., 2018), NER Model with Transformers (later on ***Transformers***) (Schweter & Akbik, 2020), Zero-shot NER with TARS (later on ***TARS***) (Halder et al., 2020).

The ***Flair Embeddings*** model uses stacked embeddings, i.g. combination of contextual string embeddings with GloVe (static embeddings model) (Pennington et al., 2014). Contextual string embeddings were proposed by Akbik et al. (2018). This approach generates different embeddings for the same word depending on its context. The ***Transformers*** model is a set of best hyperparameters to perform a NER on document level using fine-tuning or feature-based LSTM-CRF with RoBERTa (Liu et al., 2019). The ***TARS*** model allows to conduct NER without any training data or with a small dataset.

2.2. Acknowledgements corpus

As WoS contains millions of metadata records, the data chosen for the present study was restricted by year and scientific domain[4]. Records from four different scientific domains published from 2014 to 2019 were considered: two domains from the social sciences (**sociology** and **economics**) and **oceanography** and **computer science** for comparison. Only WoS records types "*article*" and "*review*", published in a scientific journal in English were selected[5].

The acknowledgments corpus should be restricted to approximately 200,000 entries. Approximately 50,000 records should have been taken from each scientific domain (exact numbers are in the Column 3 of Table 1), which resulted in the total number of records in the acknowledgments corpus of 198,022 entries.

1	**2**	**3**
Scientific domain	**Total number of records**	**Number of records in the acknowledgments corpus**
oceanography	217,710	49,782
economics	145,720	49,616
computer science	962,246	49,133
sociology	497,999	49,491
total	**1,325,676**	**198,022**

Table 1: Total numbers of records stored in WoS and published between 2014 and 2019 in English with acknowledgments for each scientific domain and number of articles selected for the analyzed acknowledgment corpus.

Each scientific domain in WoS consists of several disciplines. For example, domain Economics includes the following disciplines: Economics, Agricultural Economics & Policy, and Business & Economics. Entries from each discipline should have been presented in the acknowledgement copus. Therefore, the approximate number of records to be selected from

[4] List of WoS disciplines: https://github.com/kalawinka/minack/blob/results/wos_disciplines_full.csv

[5] List of the selected disciplines and the number of records for each discipline: https://github.com/kalawinka/minack/blob/results/counts_disciplines_single_discipline.csv

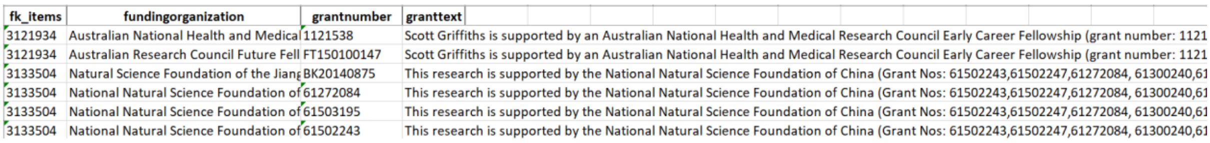
each discipline was calculated[6]. This caused slight differences in numbers of records in different disciplines.

2.3. Training corpora[7]

A creation of the training corpora was conducted in three steps. At the first step, 1000 acknowledgments texts were gathered from the WoS. Choosing criteria are similar to ones used for the acknowledgments corpus. Additionally, only articles containing indexed funding organisations and grant numbers were selected. WoS entries were restricted according to the described above choosing criteria. Further, the first 1000 distinct entries were retrieved for the training corpus. At the second step, training data were annotated. At the third step, the resulting corpus was divided into two corpora of different sizes.

2.3.1. Step: Annotation

As already mentioned, WoS contains indexed funding information. Figure 2 demonstrates an example of funding information indexed in WoS.



fk_items	fundingorganization	grantnumber	granttext
3121934	Australian National Health and Medical	1121538	Scott Griffiths is supported by an Australian National Health and Medical Research Council Early Career Fellowship (grant number: 1121
3121934	Australian Research Council Future Fell	FT150100147	Scott Griffiths is supported by an Australian National Health and Medical Research Council Early Career Fellowship (grant number: 1121
3133504	Natural Science Foundation of the Jiang	BK20140875	This research is supported by the National Natural Science Foundation of China (Grant Nos: 61502243,61502247,61272084, 61300240,61
3133504	National Natural Science Foundation of	61272084	This research is supported by the National Natural Science Foundation of China (Grant Nos: 61502243,61502247,61272084, 61300240,61
3133504	National Natural Science Foundation of	61503195	This research is supported by the National Natural Science Foundation of China (Grant Nos: 61502243,61502247,61272084, 61300240,61
3133504	National Natural Science Foundation of	61502243	This research is supported by the National Natural Science Foundation of China (Grant Nos: 61502243,61502247,61272084, 61300240,61

Figure 2: Example of funding information indexed in WoS.

WoS funding information indexing has several issues. Not every acknowledgment text has indexed funding information. Only funding information is included, i.e. individuals are not indexed. Indexed funding organisations are not divided into different entity types like universities, corporations, etc. Existing indexing of funding organisations is incomplete, as Table 2 demonstrates.

1	**2**	**3**
Acknowledgment text	**Entities indexed in WoS**	**Not indexed entities**
Support for this work was provided in part by the National Institute of Mental Health (R01 MH071589 to LP.) and a fellowship of the Japan Society for the Promotion of	● National Institute of Mental Health ● R01 MH071589	● Japan Society for the Promotion of Science ● Dr. Steven Most

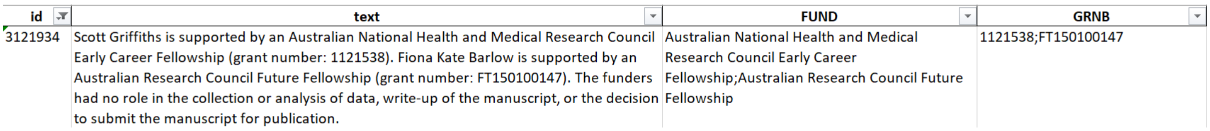
[6] Number of disciplines in each scientific domain and the number of records in the corpus for each scientific domain: https://github.com/kalawinka/minack/blob/results/counts_disciplines_total.csv

[7] Full annotated training corpus (1000 records): https://github.com/kalawinka/minack/blob/results/annotated_corpus_1000.csv

Science (to TY). We thank Dr. Steven Most for sharing pictures from his previous study. We also thank Brenton McMenamin and Jong Moon Choi for discussions of this work.		- Brenton McMenamin - Jong Moon Choi

Table 2: Example of WoS indexing problems. Acknowledged entities are marked with different colours according to classification provided in Figure 1 (colours in Table 2 match colours in Figure 1). Column 1 contains a sentence from an acknowledgment. Column 2 demonstrates which acknowledged entities from that sentence are indexed in WoS. Column 3 shows what entities are absent in the WoS indexing.

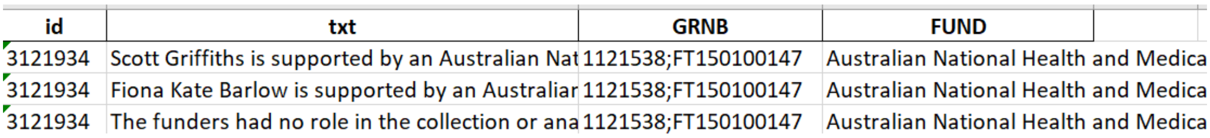
For the corpus annotation, a semi-automatic approach was developed. Firstly, the corpus design was adjusted to the less redundant format. Indexed funding organisations, grant numbers and texts were merged into one row by text id. Duplicated entities within one acknowledgement were deleted. Figure 3 shows the example of merged funding information indexed in WoS.



id	text	FUND	GRNB
3121934	Scott Griffiths is supported by an Australian National Health and Medical Research Council Early Career Fellowship (grant number: 1121538). Fiona Kate Barlow is supported by an Australian Research Council Future Fellowship (grant number: FT150100147). The funders had no role in the collection or analysis of data, write-up of the manuscript, or the decision to submit the manuscript for publication.	Australian National Health and Medical Research Council Early Career Fellowship;Australian Research Council Future Fellowship	1121538;FT150100147

Figure 3: Example of merged funding information indexed in WoS.

Further, all acknowledgement texts were splitted into single sentences[8], using segtok.segmenter[9] (GitHub, 2020), as Figure 4 demonstrates. Splitting of sentences was manually examined and corrected.



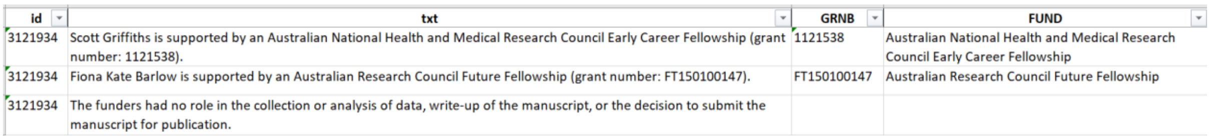
id	txt	GRNB	FUND
3121934	Scott Griffiths is supported by an Australian Nat	1121538;FT150100147	Australian National Health and Medica
3121934	Fiona Kate Barlow is supported by an Australiar	1121538;FT150100147	Australian National Health and Medica
3121934	The funders had no role in the collection or ana	1121538;FT150100147	Australian National Health and Medica

Figure 4: Example of funding information indexed in WoS in the splitted by sentence format.

In the next step redundant indexing was eliminated, as Figure 5 demonstrates. Only entities that are present in the sentence were left in the annotation columns. Entity matching was conducted using regular expressions (Python Software Foundation, 2021).

[8] Flair annotation format requires a text in a corpus to be divided into single sentences.

[9] Flair uses segtok.segmenter to divide analysed texts into sentences.

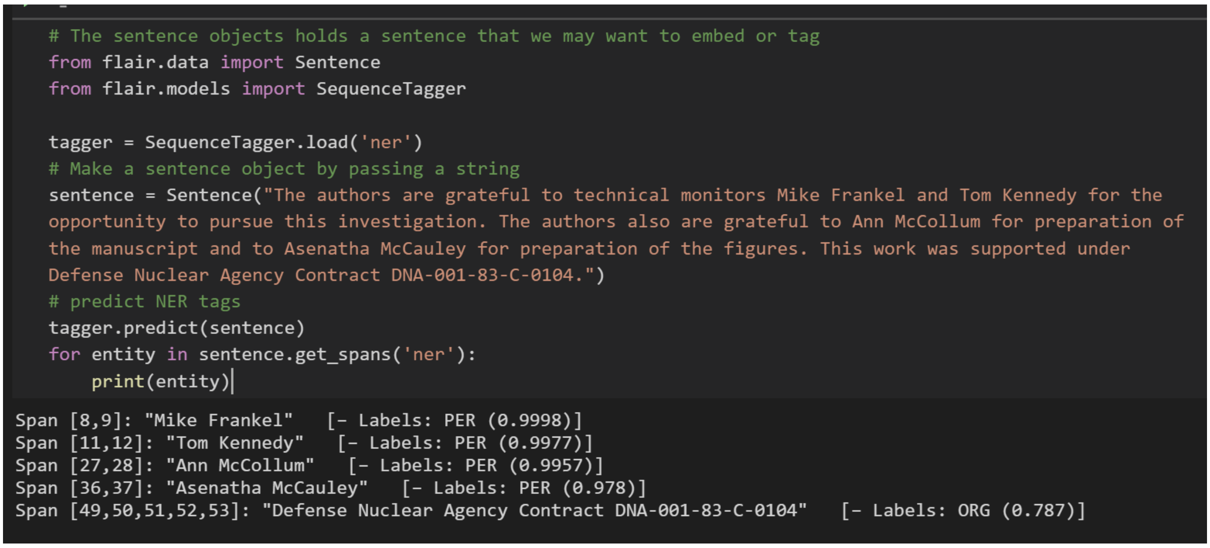


id	txt	GRNB	FUND
3121934	Scott Griffiths is supported by an Australian National Health and Medical Research Council Early Career Fellowship (grant number: 1121538).	1121538	Australian National Health and Medical Research Council Early Career Fellowship
3121934	Fiona Kate Barlow is supported by an Australian Research Council Future Fellowship (grant number: FT150100147).	FT150100147	Australian Research Council Future Fellowship
3121934	The funders had no role in the collection or analysis of data, write-up of the manuscript, or the decision to submit the manuscript for publication.		

Figure 5: Example of not redundant funding information indexed in WoS in the splitted by sentence format.

The training corpus should have been annotated with six types of entities. Some of the entities were already completely (i.g. grant numbers) or partly (i.g. funding organisation) indexed in WoS. Therefore, grant numbers were adopted from the WoS indexing unaltered.

FLAIR has a pretrained 4-class NER FLAIR model (CoNLL-03) (Akbik, 2021). The model is able to predict four tags: PER (person name), LOC (location), ORG (organisation name) and MISC (other name). Figure 2 demonstrates the example of using pretrained 4-class NER FLAIR model (CoNLL-03) on the acknowledgement text.



```
# The sentence objects holds a sentence that we may want to embed or tag
from flair.data import Sentence
from flair.models import SequenceTagger

tagger = SequenceTagger.load('ner')
# Make a sentence object by passing a string
sentence = Sentence("The authors are grateful to technical monitors Mike Frankel and Tom Kennedy for the
opportunity to pursue this investigation. The authors also are grateful to Ann McCollum for preparation of
the manuscript and to Asenatha McCauley for preparation of the figures. This work was supported under
Defense Nuclear Agency Contract DNA-001-83-C-0104.")
# predict NER tags
tagger.predict(sentence)
for entity in sentence.get_spans('ner'):
    print(entity)
```

```
Span [8,9]: "Mike Frankel"   [− Labels: PER (0.9998)]
Span [11,12]: "Tom Kennedy"   [− Labels: PER (0.9977)]
Span [27,28]: "Ann McCollum"   [− Labels: PER (0.9957)]
Span [36,37]: "Asenatha McCauley"   [− Labels: PER (0.978)]
Span [49,50,51,52,53]: "Defense Nuclear Agency Contract DNA-001-83-C-0104"   [− Labels: ORG (0.787)]
```

Figure 6: Output of the pretrained 4-class CoNLL-03 FLAIR model.

As FLAIR showed adequate results in extraction of names of individuals, it was decided to apply the pretrained 4-class CoNLL-03 FLAIR model to the training dataset. Entities which fell into the PER category were added as the IND annotation to the training corpus. Besides, we noticed that some funding information was partially correctly extracted into the ORG and MISC categories[10]. Therefore, WoS funding organisation indexing and entities from the ORG and MISC categories were adopted and distinguished between three categories (FUND, COR and UNI) using regular expressions: recognized entities were ascribed to the three categories

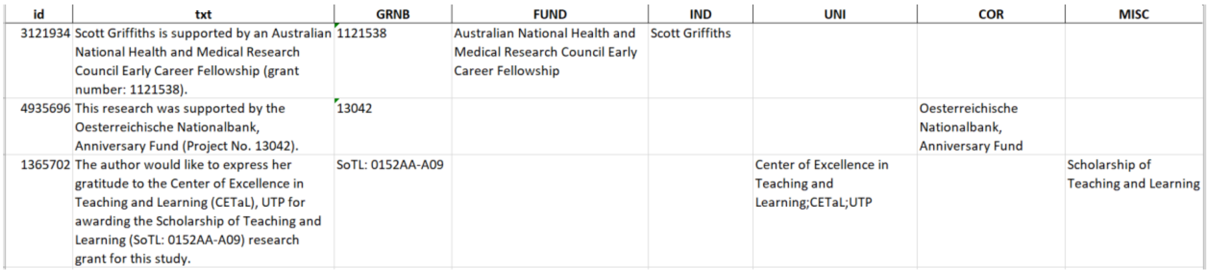
[10] Full 4-class CoNLL-03 FLAIR output: https://github.com/kalawinka/minack/blob/results/4-class_CoNLL-03_FLAIR_output.csv

using the python re library (Python Software Foundation, 2021) according to the specific patterns. Entries containing specific patterns (Table 3, Column 3) as substring were ascribed to FUND or UNI categories. Entries, for which no pattern matches were found, were ascribed to the COR category. Patterns were defined according to Giles and Council classification (2004, p. 17601).

1	**2**	**3**
Category	**Category abbreviation**	**Pattern**
funding agencies	(FUND)	foundation; agency; research; department; academy; fund; programme; capitalized abbreviations; project; ministe*r; government; european union; national; fond; laboratory, cente*r, study, society, trust, science, fellowship; grant; hospital
universities	(UNI)	universit; institute (not preceded by national)

Table 3: Acknowledged entities patterns.

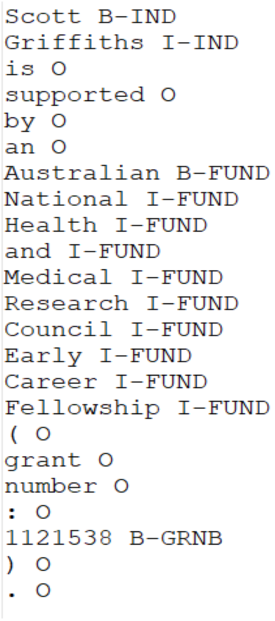
Results of the automatic annotation were saved as a table in the excel format. Further, automatic classification of entities was manually examined and reviewed. Category's mismatching, not completely extracted entities and not extracted entities were corrected. Acknowledged entities, which fall into the MISC category, were annotated manually. We believe the semi-automatic approach to be more time-saving than complete manual annotation. Figure 7 demonstrates the final form of the annotated training corpus. We chose the excel table format, as this format was convenient for annotation examination and correction.



id	txt	GRNB	FUND	IND	UNI	COR	MISC
3121934	Scott Griffiths is supported by an Australian National Health and Medical Research Council Early Career Fellowship (grant number: 1121538).	1121538	Australian National Health and Medical Research Council Early Career Fellowship	Scott Griffiths			
4935696	This research was supported by the Oesterreichische Nationalbank, Anniversary Fund (Project No. 13042).	13042				Oesterreichische Nationalbank, Anniversary Fund	
1365702	The author would like to express her gratitude to the Center of Excellence in Teaching and Learning (CETaL), UTP for awarding the Scholarship of Teaching and Learning (SoTL: 0152AA-A09) research grant for this study.	SoTL: 0152AA-A09			Center of Excellence in Teaching and Learning;CETaL;UTP		Scholarship of Teaching and Learning

Figure 7: Example of the final corpus annotation in the excel format.

A training corpus for the FLAIR model should be in a specific annotation format, which is shown in Figure 8. At the last annotation step, the corpus in excel format was converted to the FLAIR format. We used the IOB2-format for tag annotation ('Inside–Outside–Beginning (Tagging)', 2021). Words marked B- indicate the beginning of the annotated chunk, words marked I- are inside the annotated chunk and words marked O are outside the annotated chunk.



```
Scott B-IND
Griffiths I-IND
is O
supported O
by O
an O
Australian B-FUND
National I-FUND
Health I-FUND
and I-FUND
Medical I-FUND
Research I-FUND
Council I-FUND
Early I-FUND
Career I-FUND
Fellowship I-FUND
( O
grant O
number O
: O
1121538 B-GRNB
) O
. O
```

Figure 8: Example of the FLAIR column format.

2.3.2. Step: Two training corpora

In order to train the FLAIR custom NER Tagger model, two corpora containing 49[11] and 654[12] acknowledgment texts were created. The effectiveness of corpora of different sizes was tested in order to find out the most efficient training corpus size. The training corpora consist of a training set (train), a test set (test) and a validation set (dev). Table 3 demonstrates the amount of sentences in each set in two corpora.

Corpus No.	**Training set (train)**	**Test set (test)**	**Validation set (dev)**
1	29	10	10
2	339	165	150

Table 4: Number of sentences in the training corpora.

[11] Training corpora no.1 in csv format: https://github.com/kalawinka/minack/blob/results/train_small_no1.csv; https://github.com/kalawinka/minack/blob/results/test_small_no1.csv; https://github.com/kalawinka/minack/blob/results/dev_small_no1.csv;
Training corpora no.1 in IOB2 format: https://github.com/kalawinka/minack/blob/results/train_small.txt; https://github.com/kalawinka/minack/blob/results/test_small.txt; https://github.com/kalawinka/minack/blob/results/dev_small.txt;

[12] Training corpora no.2 in csv format: https://github.com/kalawinka/minack/blob/results/train_big_no2.csv; https://github.com/kalawinka/minack/blob/results/test_big_no2.csv; https://github.com/kalawinka/minack/blob/results/dev_big_no2.csv;
Training corpora no.2 in IOB2 format: https://github.com/kalawinka/minack/blob/results/train_big.txt; https://github.com/kalawinka/minack/blob/results/test_big.txt; https://github.com/kalawinka/minack/blob/results/dev_big.txt;

As WoS only stores acknowledgement texts, which contains funding information, there was a disproportion between occurrences of entities of different types.

Category	**Number of sentences, containing category**
FUND	1209
GRNB	1119
IND	476
UNI	306
MISC	237
COR	42

Table 5: Distribution of sentences containing acknowledged entities of different types.

As Table 5 demonstrates, GRNB and FUND were the most represented categories and COR was the least represented category. We tried to pick an equal number of sentences with each entity category in order to make a well balanced corpus. However, that was impossible for the COR category, due to the limited number of sentences containing this category. That way all entries of the COR category were selected for the Corpus No. 2, as Figure 10 demonstrates.

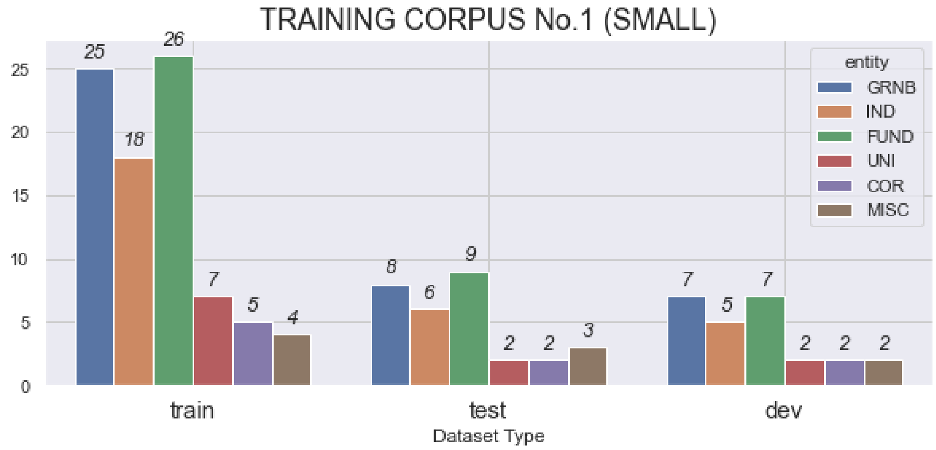


Figure 9: Distribution of sentences with acknowledged entities of each type in the training dataset No.1.

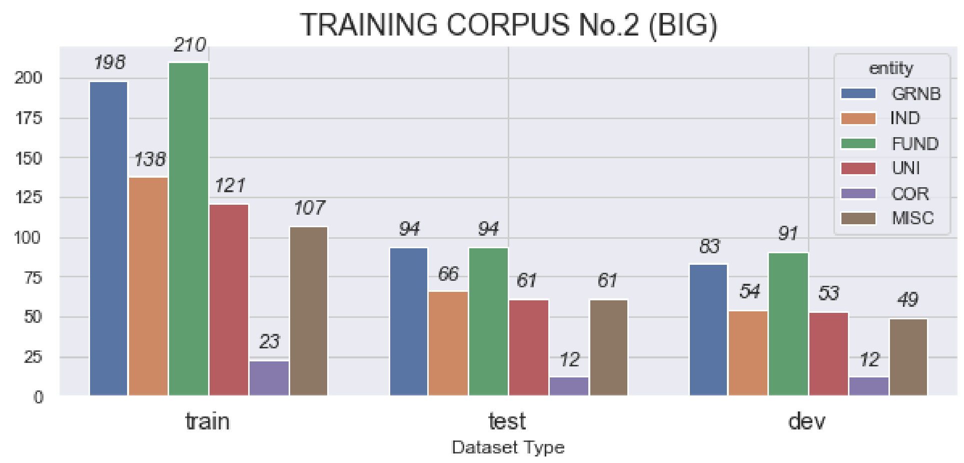


Figure 10: Distribution of sentences with acknowledged entities of each type in the training dataset No.2.

Next important criterion was that all sets (train, test and dev) should have had sentences with entities of each type. Figures 9 and 10 demonstrate distribution of sentences containing different types of acknowledged entities in the training corpora.

3. Results

3.1. Primary training[13]

Primary training was conducted using three default FLAIR training algorithms described in section 2.1. Firstly, training was performed with the dataset no. 1 (small dataset). Figure 11 demonstrates the results of the training with the dataset no. 1.

Overall training demonstrated mixed results. IND and GRNB showed adequate results by training with Flair Embeddings and TARS. IND was the best recognized entity by training with Flair Embeddings and TARS with a f1-score of 0,8 (Flair Embeddings) and 0,8571 (Tars). Training with Transformers was not successful for IND with a f1-score of 0. Transformers averall proved to be a less efficient algorithm for training with the small dataset, with the overall accuracy of 0.3485 (Figure 13). FUND demonstrated not adequate results with f1-score less than 0.5 for all algorithms (Figure 11). Entity types MISC, UNI and COR showed the worst results with the f1-score equal to zero for all algorithms (Figure 11). Low accuracy for MISC, UNI and COR resulted in low overall accuracy for all algorithms (Figure 13). Overall training

[13] Results of the primary training: https://github.com/kalawinka/minack/blob/results/logs_compare_small.txt; https://github.com/kalawinka/minack/blob/results/logs_compare_big.txt; https://github.com/kalawinka/minack/blob/results/accuracy_primary_training.csv

with the dataset no. 1 showed not sufficient results for all algorithms. Flair Embeddings and TARS, though, showed better accuracy in comparison with Transformers.

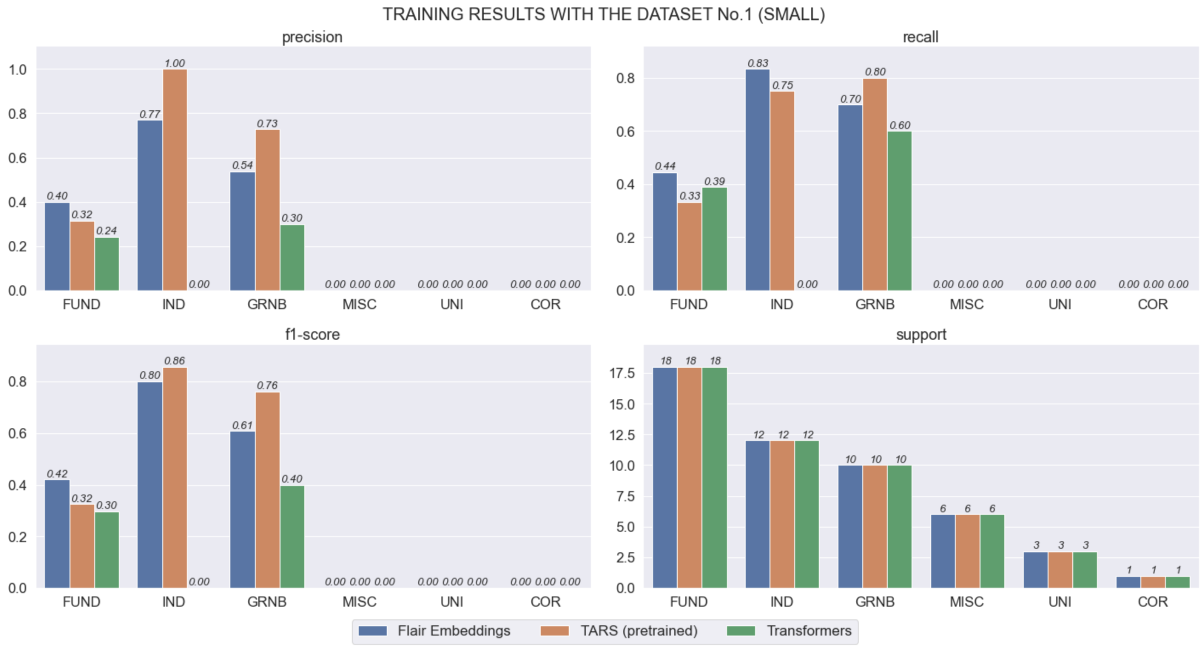


Figure 11: Training results with the training set No.1.

Further, training with the dataset no. 2 (big) was performed. Figure 12 demonstrates training results with the dataset No.2. Similar to the training with dataset no. 1 IND and GRNB are the best recognized categories. Best results for IND and GRNB demonstrated Flair Embeddings with a f1-score of 0,9797 (IND) and 0,9571 (GRNB). TARS achieved the best results for FUND with a f1-score of 0,7651, against 0,7093 for Flair Embeddings and 0,6801 for Transformers. Miscellaneous demonstrated the worst accuracy for Flair Embeddings (0,638) and Transformers (0,4881), while for TARS the worst accuracy lies by COR category with a f1-score of 0,5385. Best result for UNI showed Flair Embeddings with a f1-score over 0,7.

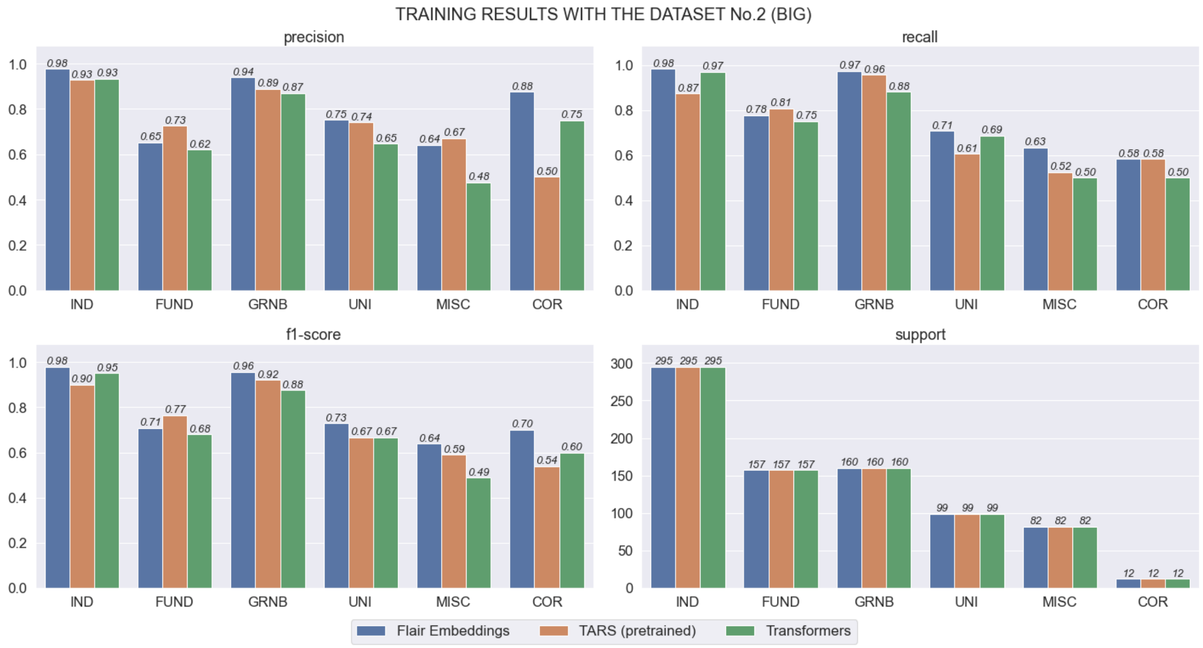


Figure 12: Training results with the training set No.2.

Training with dataset no.2 showed extreme improvement in training accuracy (Figure 13). Overall, Flair Embeddings was more accurate than other training algorithms, although training with TARS showed better results for the FUND category. Transformers surprisingly showed the worst results during the training.

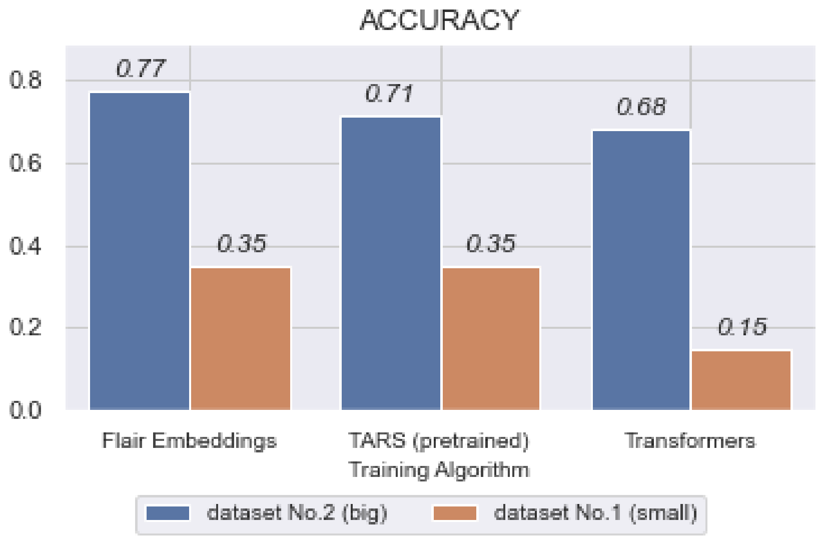


Figure 13: Accuracy of training algorithms.

As Flair Embeddings showed the highest overall accuracy of 0.7702, it was decided to conduct analysis with the model trained with this algorithm.

3.2. Additional training[14]

In order to understand the reasons for the low accuracy of some entity types (FUND, COR, MISC, UNI) and in hope to improve the results we decided to conduct some additional experiments.

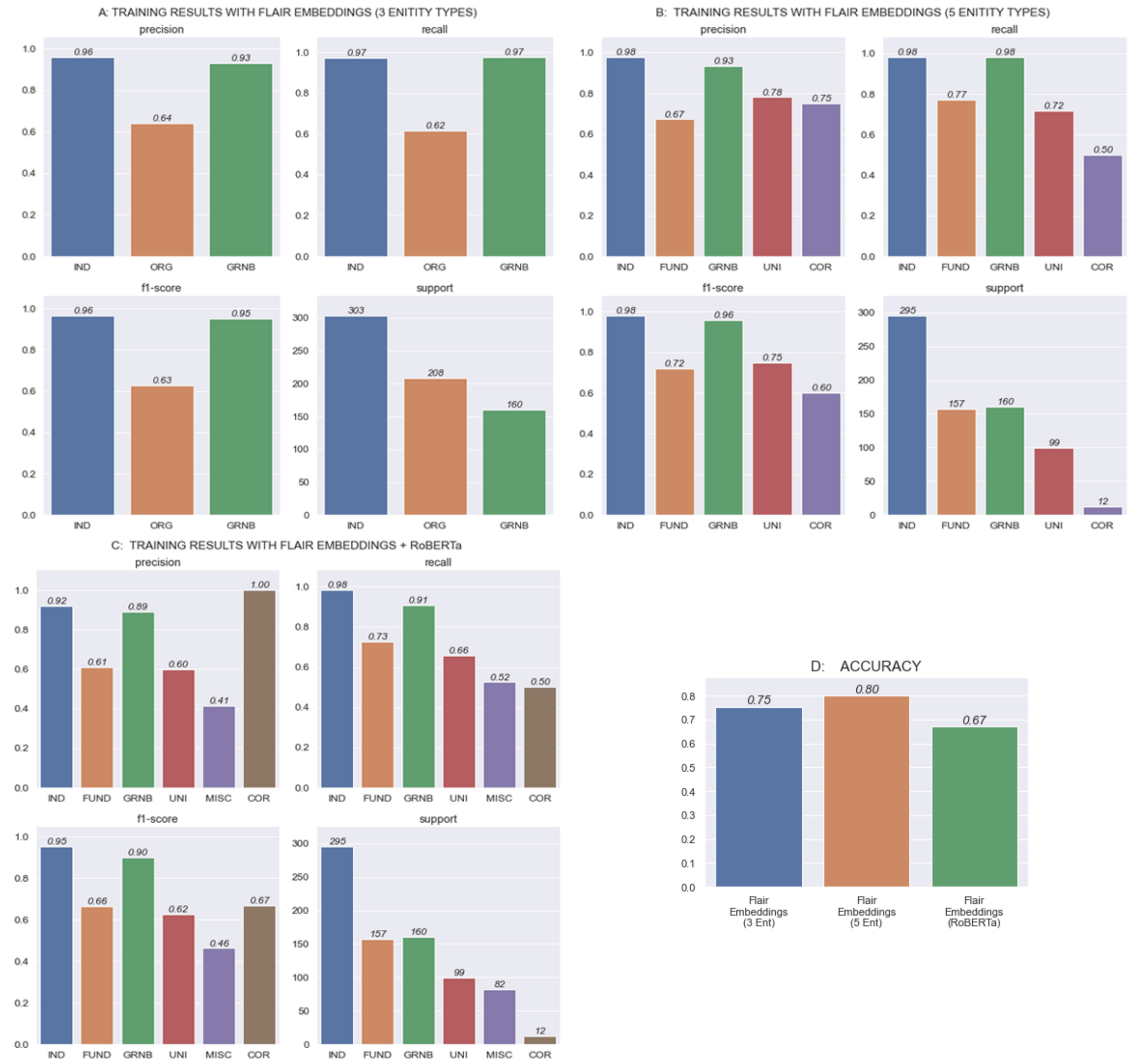


Figure 14: results of the additional training.

Our first hypothesis was that these four categories (FUND, COR, MISC, UNI) are very close semantically, which prevents successful recognition. To examine that theory we conducted an experiment using Flair Embedding with the dataset containing entities of 3 types: IND, GRNB

[14] Results of the additional training: https://github.com/kalawinka/minack/blob/results/comparison_add_training.csv; https://github.com/kalawinka/minack/blob/results/accuracy_add_training.csv;

and ORG. ORG includes a combination of entities from the FUND, COR and UNI categories. Results of the training are represented in Figure 14-A. IND and GRNB still achieved high f1-scores of 0.9639 (IND) and 0.9512 (GRNB). Nevertheless, ORG gained only a f1-score of 0,64, which is worse than previous results with six entity types.

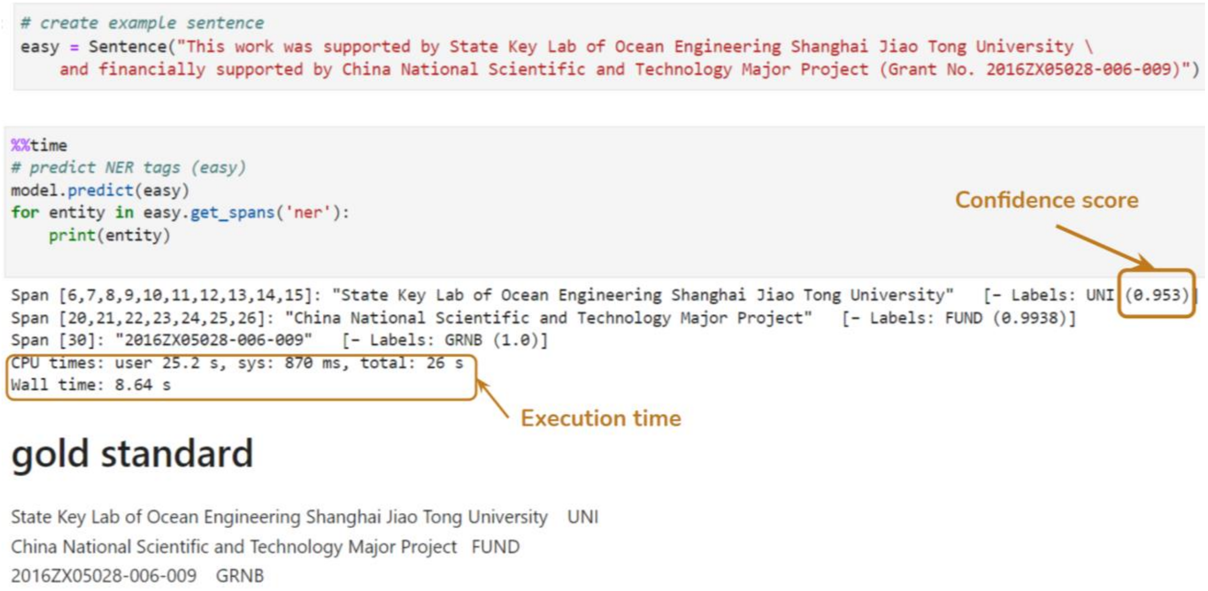
Low results for FUND, COR, MISC and UNI categories might also lie in the nature of the miscellaneous category, as some entities that fall into this category are semantically very close to FUND and COR categories. For that reason we conducted training with Flair Embeddings with a dataset excluding the MISC category, i.e. with five entity types. Training results are shown in Figure 14-B. Training results were quite similar to those achieved during the training with the dataset with six entity types. Improvement in overall accuracy (Figure 14-D) (0.799 vs. previous result of 0.7702) could be explained by the fact that MISC was not present in this training and could not affect the overall accuracy with its low f1-score.

In the third experiment wie slightly changed the training algorithm. FLAIR creators claimed Transformers to be the most successful algorithm for the NER task (Schweter & Akbik, 2020), although in our training Transformers showed the weakest accuracy. Additionally, stacked embeddings showed better performance as pure contextual string embeddings (Akbik et al., 2018, p. 1644). Therefore, for the third additional training we combined contextual string embeddings with RoBERTa (vs. contextual string embeddings + GloVe in primary training). Training results are represented in Figure 14-C. The proposed method showed no improvements compared to the results of the primary training with Transformers.

4. Results of acknowledgements NER with the best model[15]

A model with the highest accuracy (Flair Embeddings) was applied to conduct a NER on the acknowledgement corpus. As Figure 15 demonstrates the model is able to successfully recognize and label acknowledged entities in a simple sentence. Automatically annotated entities match the gold standard.

[15] Model output: https://gesisbox.gesis.org/index.php/s/e8fTos84Wf2fmje



```
# create example sentence
easy = Sentence("This work was supported by State Key Lab of Ocean Engineering Shanghai Jiao Tong University \
    and financially supported by China National Scientific and Technology Major Project (Grant No. 2016ZX05028-006-009)")
```

```
%%time
# predict NER tags (easy)
model.predict(easy)
for entity in easy.get_spans('ner'):
    print(entity)
```

```
Span [6,7,8,9,10,11,12,13,14,15]: "State Key Lab of Ocean Engineering Shanghai Jiao Tong University"   [- Labels: UNI (0.953)]
Span [20,21,22,23,24,25,26]: "China National Scientific and Technology Major Project"   [- Labels: FUND (0.9938)]
Span [30]: "2016ZX05028-006-009"   [- Labels: GRNB (1.0)]
CPU times: user 25.2 s, sys: 870 ms, total: 26 s
Wall time: 8.64 s
```

gold standard

```
State Key Lab of Ocean Engineering Shanghai Jiao Tong University   UNI
China National Scientific and Technology Major Project   FUND
2016ZX05028-006-009   GRNB
```

Figure 15: example of the FLAIR NER tagger trained with the Flair Embeddings model. In the first line we created a Sentence object from the sentence: "This work was supported by State Key Lab of Ocean Engineering Shanghai Jiao Tong University and financially supported by China National Scientific and Technology Major Project (Grant No. 2016ZX05028-006-009)." The second line generates spans with labelled acknowledged entities from the Sentence object. The third line demonstrates a gold standard: manually annotated acknowledged entities.

Figure 16 demonstrates the distribution of entities of different types between scientific domains[16]. Distribution of entities shows clear differences among scientific domains. Therefore, IND is the most frequent entity type in economics, while FUND is the most frequent in social science and oceanography and GRNB in computer science. Social science and oceanography domains show similar acknowledged entities' frequency patterns for FUND, IND and GRNB (in a descending order starting from FUND). COR is the most infrequent category in all scientific domains, followed by UNI and MISC in all scientific domains except economics. In economics GRNB showed to be the rarest entity type. Computer science demonstrates the smallest amount of acknowledged individuals.

[16] Distribution of entities of different types between scientific domains: https://github.com/kalawinka/minack/blob/results/analysis_raw_labels_frequency.csv

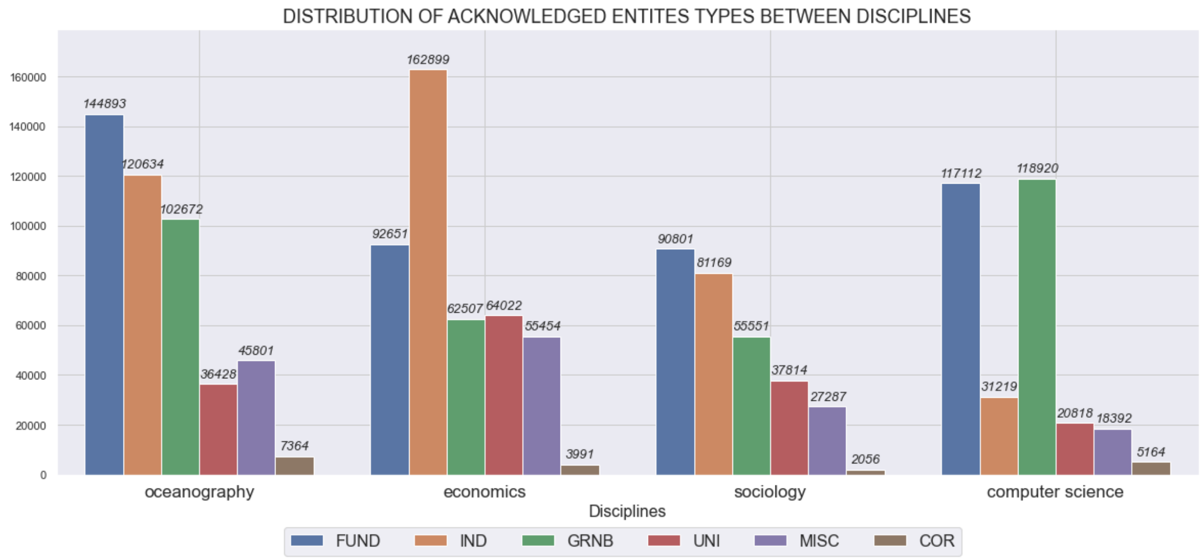


Figure 16: Distribution of acknowledged entities between scientific domains.

Figures 17 - 20 demonstrate the top 30 of acknowledged entities of different types except GRNB. Entities depicted in Figures 17 - 20 are disambiguated entities (for more details see section 5)[17]. As Figure 17 shows, all scientific domains except sociology have similar top 2 funding organisations: the National Natural Science Foundation of China and the United States-Israel Binational Science Foundation (BSF). Top 1 funding organisation for sociology is National Institutes of Health (NIH). Additionally, scientists in computer science tend to write names of individuals in abbreviated format (first letter of the name followed by surname) while in other scientific domains full format is prevailing.

[17] Analysis of disambiguated results:
https://github.com/kalawinka/minack/blob/results/analysis_disambiguated_entity_comp.csv;
https://github.com/kalawinka/minack/blob/results/analysis_disambiguated_entity_eco.csv;
https://github.com/kalawinka/minack/blob/results/analysis_disambiguated_entity_ocean.csv;
https://github.com/kalawinka/minack/blob/results/analysis_disambiguated_entity_soc.csv;
https://github.com/kalawinka/minack/blob/results/analysis_disambiguated_entity_total.csv

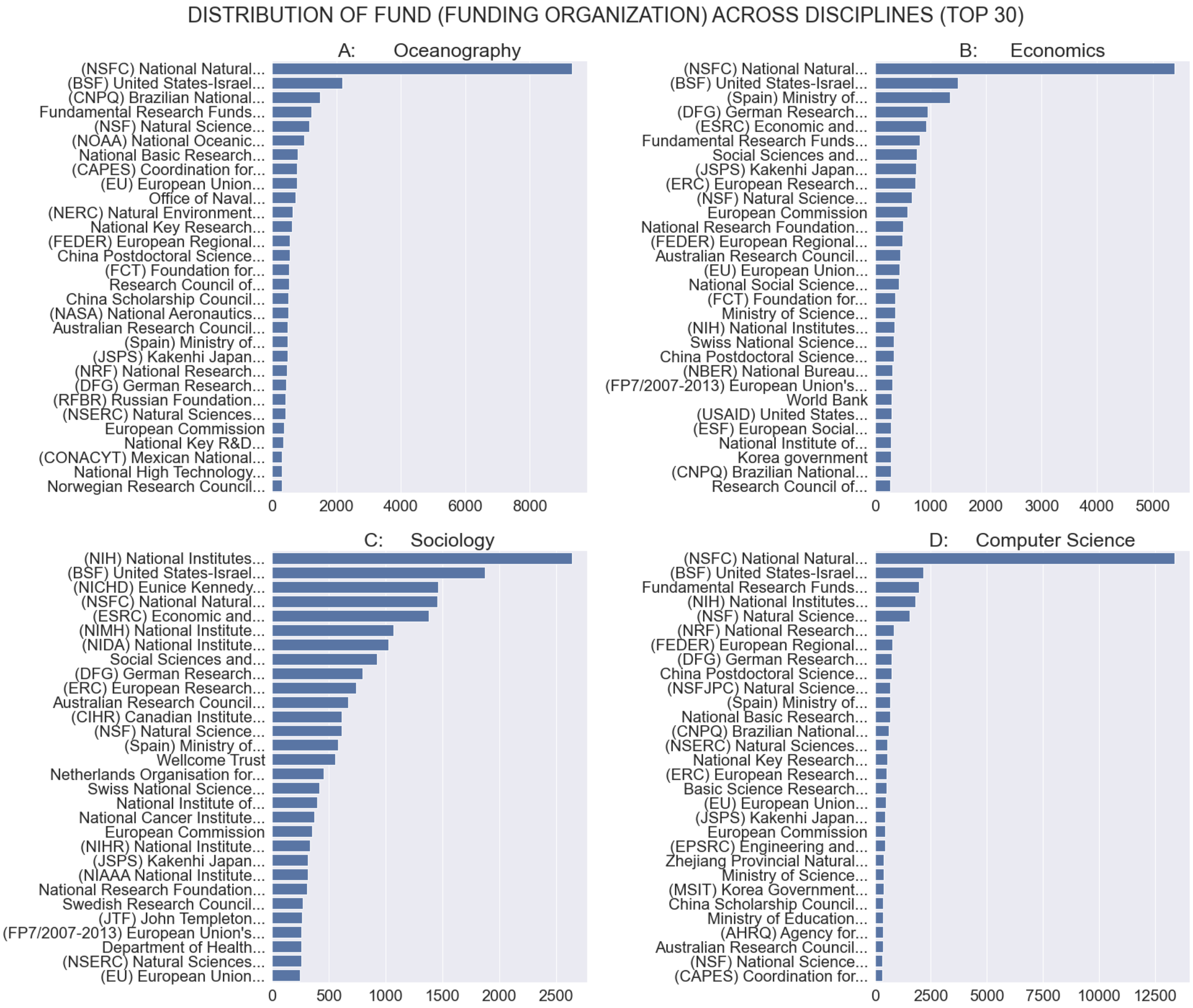


Figure 17: top 30 acknowledged entities, which fall into the FUND (funding organisation) category. Figure A represents entities from oceanography, figure B from economics, figure C from social science, and D from computer science.

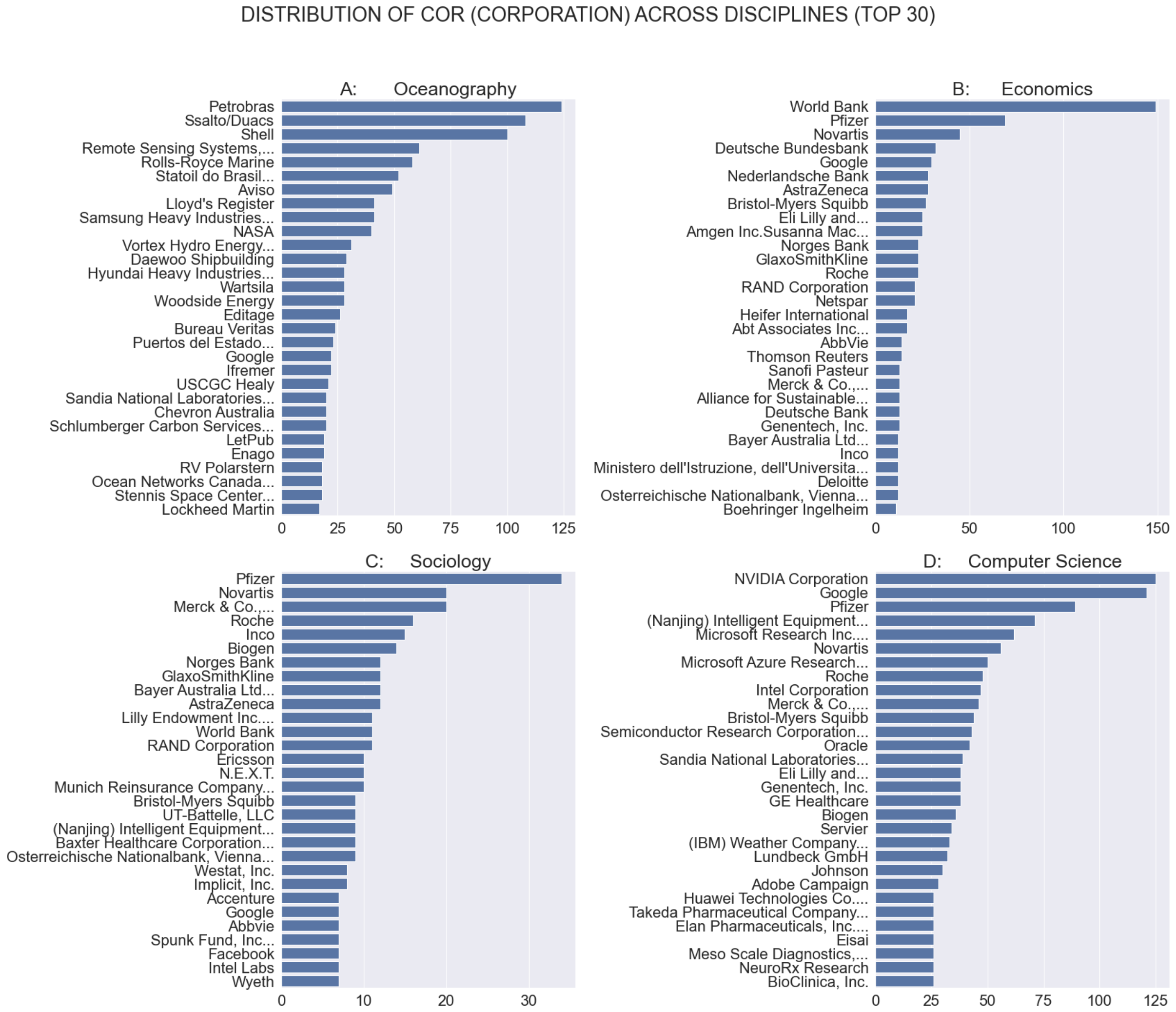


Figure 18: top 30 acknowledged entities, which fall into the COR (corporation) category. Figure A represents entities from oceanography, figure B from economics, figure C from social science, and D from computer science.

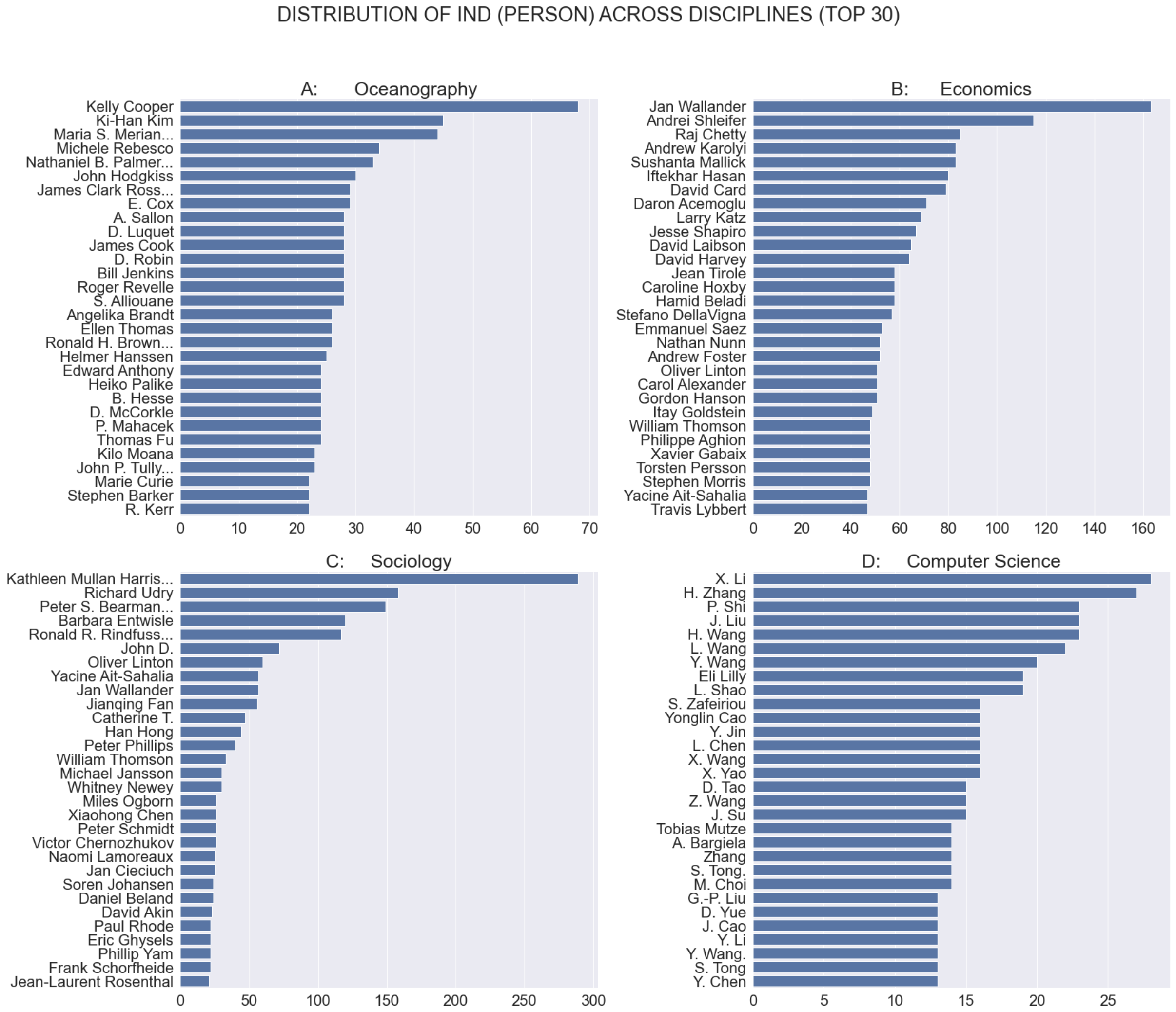


Figure 19: top 30 acknowledged entities, which fall into the IND (person) category. Figure A represents entities from oceanography, figure B from economics, figure C from social science, and D from computer science.

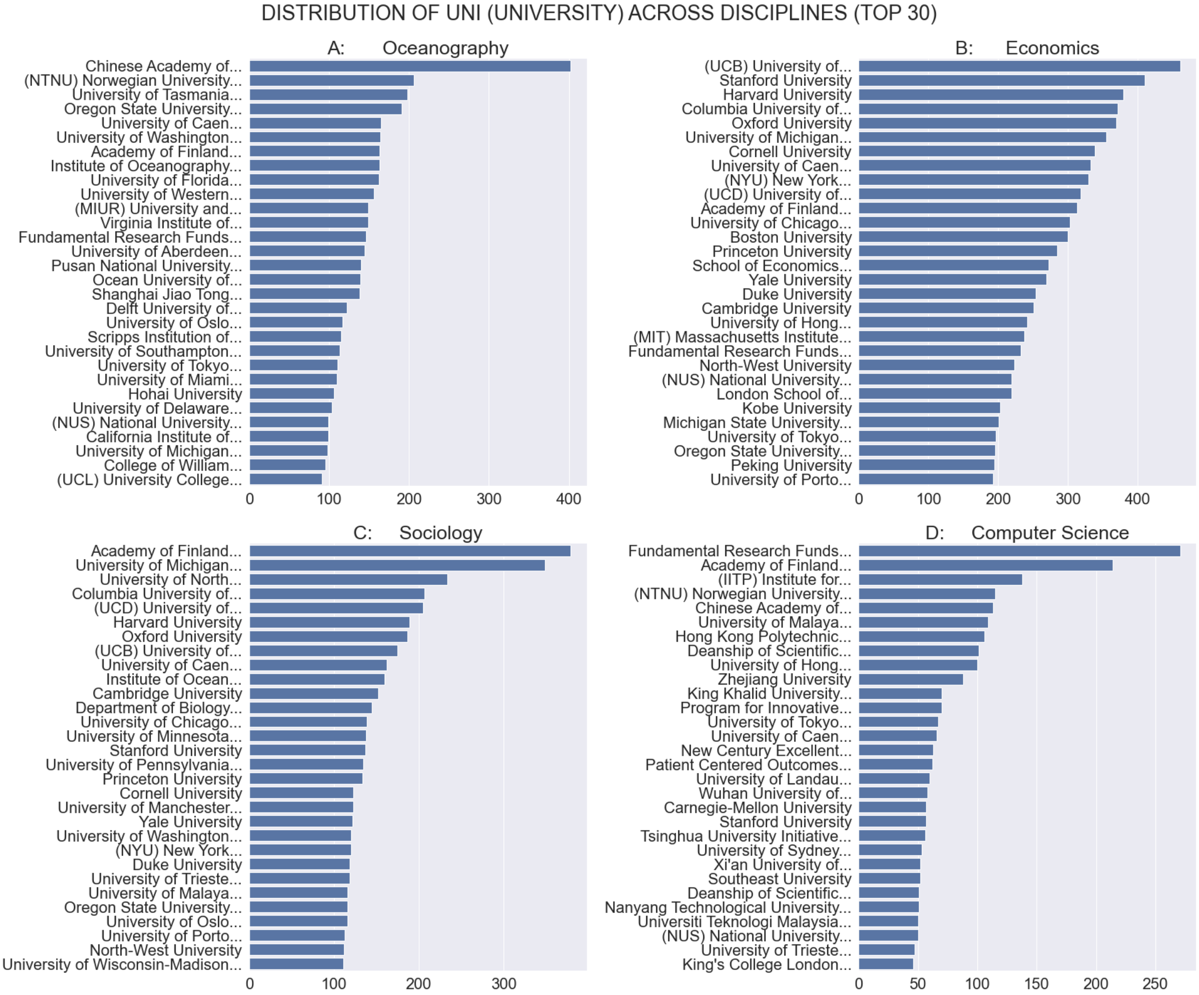


Figure 20: top 30 acknowledged entities, which fall into the UNI (university) category. Figure A represents entities from oceanography, figure B from economics, figure C from social science, and D from computer science.

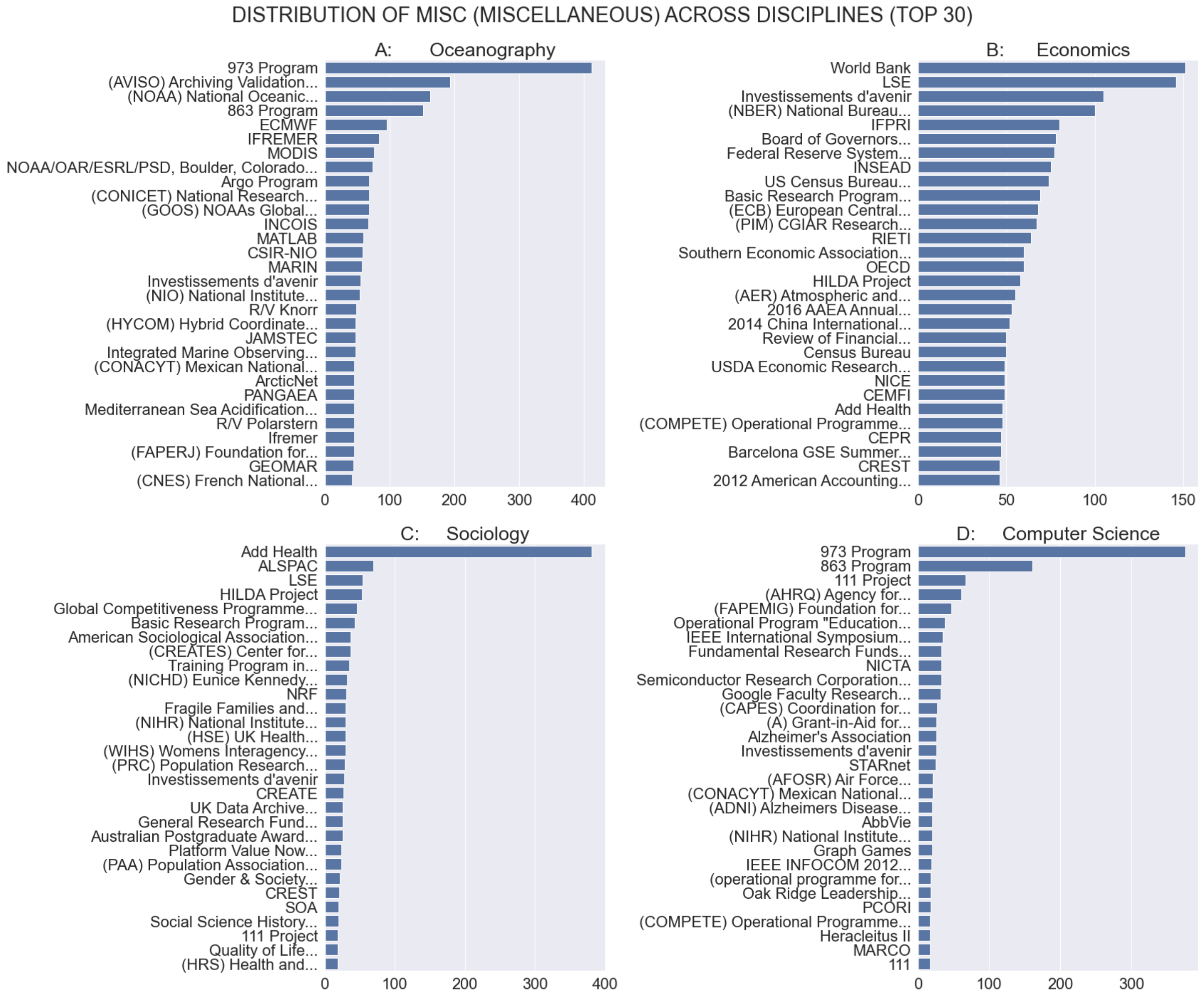


Figure 21: top 30 acknowledged entities, which fall into the MISC (miscellaneous) category. Figure A represents entities from oceanography, figure B from economics, figure C from social science, and D from computer science.

5. Challenges

NER with FLAIR showed in general adequate results, but after reviewing the first analysis of retrieved entities from the acknowledgement corpus[18] we realised that acknowledged entities should be disambiguated for plausible analysis[19]. Some entities have more than one writing variant, as Example 1 demonstrates. All variants should have been reduced to one variant.

[18] Analysis of not disambiguated results:
https://github.com/kalawinka/minack/blob/results/analysis_raw_entity_total.csv;
https://github.com/kalawinka/minack/blob/results/analysis_raw_entity_soc.csv;
https://github.com/kalawinka/minack/blob/results/analysis_raw_entity_ocean.csv;
https://github.com/kalawinka/minack/blob/results/analysis_raw_entity_eco.csv;
https://github.com/kalawinka/minack/blob/results/analysis_raw_entity_comp.csv

[19] The GRNB category was excluded from the disambiguation process, as the following disambiguation techniques do not work with number formats.

Example 1:

- *National Science Foundation*
- *NSF*
- *National Science Foundation (NSF)*

To solve this problem we created our own disambiguation datasets from extracted entities for funding organisations and universities: we used the most frequent entities[20]. All the entities in the result dataset of FUND, UNI and MISC categories were compared to the disambiguation datasets using the Levenshtein distance ('Levenshtein Distance', 2021). We used the Python fuzz.ratio function (Cohen, 2020), which calculates the Levenshtein distance similarity ratio between the two strings. Entries with the fuzz.ratio value more than 93 (that number was determined by running tests on different writing variants of different entities) were replaced with the unified writing variant (for entity in the example 1 it would be National Science Foundation (NSF)) and put into the disambiguated corpus[21]. This problem also occurred for the COR category but in this case all variants of one entity could be found using the *fuzz.partial_ratio* function (Cohen, 2020). Partial_ratio picks the shortest string from the two compared strings and matches it with all substrings of the same lengths from the second string. All the entities labelled COR were compared to each other using *fuzz.partial_ratio*. Entries with a partial ratio value greater than 96 (that number was determined by running tests on different writing variants of different COR entities) were identified as one entry.

The second revealed problem was that some entities have the same abbreviations, as example 2 demonstrates. To solve this problem we created a list of duplicated abbreviations, which are the same for different entities and excluded these abbreviations from the disambiguation dataset. That way if only abbreviation (i.g. AAS) was in the FLAIR output without its full name and it matches the list of duplicated abbreviations, the abbreviation was not altered and put in the original format into a disambiguated corpus.

[20] Disambiguation datasets:
https://github.com/kalawinka/minack/blob/results/diasmbiguation_patterns_fund.csv;
https://github.com/kalawinka/minack/blob/results/diasmbiguation_patterns_uni.csv

[21] Disambiguated corpora: https://gesisbox.gesis.org/index.php/s/GMMwNFSc9BXsT7Y;
https://gesisbox.gesis.org/index.php/s/XZNYJSWJbSP8JkG;
https://gesisbox.gesis.org/index.php/s/FXMdfFJE7D7iWDd;
https://gesisbox.gesis.org/index.php/s/3wdG58ScQMCgbYz;
https://gesisbox.gesis.org/index.php/s/9WMQz6DBaKHjott

Example 2:

Australia Awards Scholarship	*AAS*
African Academy of Sciences	*AAS*

A misspelling problem (Example 3) was faced for all entity types. To solve this, all entities were compared to each other within their entity types using Levenshtein distance. Entities with the Levenshtein distance more than 90 were identified as one category. For the IND category entities with the Levenshtein distance equal to 100 were identified as one category, as in this case only entities, which differ only in upper- and lower-case writing variants (e.g. John Doe vs. john doe) were considered as different writing variants of the same entity.

Example 3:

- *National Nature Science Foundation of China*
- *Natural National Science Foundation of China*

6. Demonstrator

You can try our NER tagger demo by following this link: https://mybinder.org/v2/gh/kalawinka/minack/main?labpath=example_model.ipynb. This demo is an interactive notebook built with the Jupyter Notebook[22] and Binder.[23] Two options are available, you can try the model with our example of acknowledgement or you can type in your own acknowledgement text. To use the demo just launch one cell after another and follow the instructions, written in the notebook.

7. References

1. Akbik, A. (n.d.). *The Flair NLP Framework*. Retrieved 21 September 2020, from https://alanakbik.github.io/flair.html
2. Akbik, A. (2021). *flairNLP/flair*. https://github.com/flairNLP/flair
3. Akbik, A., Bergmann, T., Blythe, D., Rasul, K., Schweter, S. & Vollgraf, R. (2019). *FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP* (W. Ammar, A. Louis, & N. Mostafazadeh, Eds.; pp. 54–59). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-4010
4. Akbik, A., Blythe, D. & Vollgraf, R. (2018). Contextual String Embeddings for Sequence Labeling. *2018, 27th International Conference on Computational Linguistics*, 1638--1649.

[22] https://jupyter.org/

[23] https://mybinder.org/

5. Clarivate. (2021). *LibGuides: Web of Science Core Collection: Funding Information*. https://clarivate.libguides.com/woscc/funding
6. Cohen, A. (2020). *fuzzywuzzy: Fuzzy string matching in python* (0.18.0) [Python]. https://github.com/seatgeek/fuzzywuzzy
7. Giles, C. L. & Councill, I. G. (2004). Who gets acknowledged: Measuring scientific contributions through automatic acknowledgment indexing. *Proceedings of the National Academy of Sciences*, *101*(51), 17599–17604. https://doi.org/10.1073/pnas.0407743101
8. GitHub. (2020). *segtok/README.rst at master · fnl/segtok*. https://github.com/fnl/segtok/blob/master/README.rst#b-segtoksegmenter
9. Halder, K., Akbik, A., Krapac, J. & Vollgraf, R. (2020). Task-Aware Representation of Sentences for Generic Text Classification. *Proceedings of the 28th International Conference on Computational Linguistics*, 3202–3213. https://doi.org/10.18653/v1/2020.coling-main.285
10. Inside–outside–beginning (tagging). (2021). In *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Inside%E2%80%93outside%E2%80%93beginning_(tagging)&oldid=1041803321
11. Kassirer, J. P. & Angell, M. (1991). On authorship and acknowledgments. *The New England Journal of Medicine*, *325*(21), 1510–1512. https://doi.org/10.1056/NEJM199111213252112
12. Levenshtein distance. (2021). In *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Levenshtein_distance&oldid=1055332013
13. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. & Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *ArXiv:1907.11692 [Cs]*. http://arxiv.org/abs/1907.11692
14. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., … Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. *ArXiv:1912.01703 [Cs, Stat]*. http://arxiv.org/abs/1912.01703
15. Pennington, J., Socher, R. & Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. *Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. http://www.aclweb.org/anthology/D14-1162
16. Python Software Foundation. (2021). *re — Regular expression operations — Python 3.9.6 documentation*. https://docs.python.org/3/library/re.html
17. Schweter, S. & Akbik, A. (2020). FLERT: Document-Level Features for Named Entity Recognition. *ArXiv*.
18. Web of Science Group. (2021). *Trusted publisher-independent citation database - Web of Science Group*. https://clarivate.com/webofsciencegroup/solutions/web-of-science/