Evaluation of Embedding Models for Automatic Extraction and Classification of Acknowledged Entities in Scientific Documents

Nina Smirnova[1], Philipp Mayr[1]

[1]GESIS — Leibniz Institute for the Social Sciences, Cologne, Germany

Abstract

Acknowledgments in scientific papers may give an insight into aspects of the scientific community, such as reward systems, collaboration patterns, and hidden research trends. The aim of the paper is to evaluate the performance of different embedding models for the task of automatic extraction and classification of acknowledged entities from the acknowledgment text in scientific papers. We trained and implemented a named entity recognition (NER) task using the Flair NLP-framework. The training was conducted using three default Flair NER models with two differently-sized corpora. The Flair Embeddings model trained on the larger training corpus showed the best accuracy of 0.77. Our model is able to recognize six entity types: funding agency, grant number, individuals, university, corporation and miscellaneous. The model works more precise for some entity types than the others, thus, individuals and grant numbers showed very good F1-Score over 0.9. Most of the previous works on acknowledgement analysis were limited by the manual evaluation of data and therefore by the amount of processed data. This model can be applied for the comprehensive analysis of the acknowledgement texts and may potentially make a great contribution to the field of automated acknowledgement analysis.

Keywords

NLP, NER, Web of Science, Acknowledgement, Text Mining, Flair NLP-Framework,

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"* [1, p. 1511]. Cronin and Weaver [2] ascribe an acknowledgment alongside authorship and citedness to measures of a researcher's scholarly performance: a feature that reflects the researcher's productivity and impact. Giles and Councill [3] argue that acknowledgments to individuals, in the same way as citations, may be used as a metric to measure an individual's intellectual contribution to scientific work. Acknowledgements of financial support are interesting in terms of evaluating the influence of funding agencies on academic research. Acknowledgments of technical and instrumental support may reveal *"indirect contributions of research laboratories and universities to research activities"* [3, p. 17599].

The analysis of acknowledgments is particularly interesting as acknowledgments may give an insight into aspects of the scientific community, such as reward systems, collaboration patterns, and hidden research trends. From the linguistic point of view, acknowledgements are unstructured text data, which through automatic analysis poses research and methodological problems like data cleaning, choosing the right tokenization method, and whether and how word embeddings may enhance their automatic analysis.

To our knowledge, previous works on automatic acknowledgment analysis were mostly concerned with the extraction of funding organizations and grant numbers [4, 5] or classification of acknowledgement texts [6]. Furthermore, large bibliographic databases such as Web of Science (WoS)[1] and Scopus selectively index only funding information, i.e., names of funding organizations and grant identification numbers. Consequently, we want to extend that to other types of acknowledged entities.

The aim of the present paper is to evaluate the performance of existing embedding models for the task of automatic extraction and classification of acknowledged entities from the acknowledgment text in scientific papers. The Flair - an open-source Natural Language Processing (NLP) Framework [7] is used in our study for creating a tool for extraction of acknowledged entities because this library is easily customizable. It offers the possibility of creating a customized Named Entity Recognition (NER) tagger, which can be used for processing and analyzing the acknowledgement texts. Furthermore, Flair has shown better accuracy for NER tasks using pre-trained datasets in comparison with many other open

[1]http://wokinfo.com/products_tools/multidisciplinary/ webofscience/fundingsearch/

source NLP tools[2].

We trained and implemented a NER task using three default Flair NER models with two differently-sized corpora[3]. Models were trained to recognize six types of acknowledged entities: funding agency, grant number, individuals, university, corporation and miscellaneous. The model with the best accuracy can be applied for the comprehensive analysis of the acknowledgement texts. We performed an additional training with altered training parameters or altered training corpora (Section 5). Most of the previous works on acknowledgement analysis were limited by the manual evaluation of data and therefore by the amount of processed data [3, 8, 9, 10]. Furthermore, Thomer and Weber [11] argue that using of named entities can benefit the process of manual document classification and evaluation of the data. Therefore, a model, which is capable of extracting and classification of different entity types may potentially make a great contribution to the field of automated acknowledgement analysis.

Research questions

In this paper, we address the following research questions:

- **RQ1:** Which of the Flair default NER models is more suitable for the defined task of the extraction and classification of acknowledged entities from scientific acknowledgements?
- **RQ2:** How does the training corpus size impact the training accuracy for different NER models?

2. Background and Related work

Named Entity Recognition (NER) is a form of NLP, which aims to extract named entities from an unstructured text and classify them into predefined categories. A named entity is a real-world object that is important for understanding the text [12]. As a rule NER tasks require training data, i.e., a particular dataset or corpus, which is usually divided into several datasets: training set, test set and validation set. NER models require corpora with semantic annotation, i.e., metadata about concepts attached to the unstructured text data [13]. The annotation process is crucial as insufficient or redundant metadata can slow down and bias a learning process [14, Chapter 1].

We are aware of several works on automated information extraction from acknowledgements. Giles and Councill [3] developed an automated method for acknowledgment extraction and analysis using regular expressions

and the support vector machines (SVM) classification algorithm. Computer science research papers from the CiteSeer digital library were used as the data source. Extracted entities were analysed and manually assigned to the following four categories: funding agencies, corporations, universities, and individuals.

Thomer and Weber [11] used the 4 class Stanford Entity Recognizer [15] to extract persons, locations, organizations and miscellaneous entities from the collection of the bioinformatics texts from PubMed Central's Open Access corpus. Aim of the study was to determine an approach to *"increase the speed of ... classification without sacrificing accuracy, nor reliability"* [11, p. 1134].

Kayal et al. [5] introduced a method for extraction of funding organizations and grants from acknowledgement texts using a combination of sequential learning models: conditional random fields (CRF), hidden markov models (HMM) and maximum entropy models (MaxEnt). The final model contained pooling outputs of the single used models.

Alexandera and de Vries [4] proposed AckNER, a tool for financial information extraction from the funding or acknowledgment section of a research article. AckNER works with the use of dependency parse trees and regular expressions and is able to extract names of the organisations, projects, programs and funds, as also numbers of contracts and grants [4].

The Flair NLP Framework

Flair is an open-sourced NLP framework built on PyTorch [16], which is an open source machine learning library. *"The core idea of the framework is to present a simple, unified interface for conceptually very different types of word and document embeddings"* [7, p. 54]. Flair has three default training algorithms for NER which were used for primary training in the present research: a) NER Model with Flair Embeddings (later on Flair Embeddings) [17], b) NER Model with Transformers (later on Transformers) [18], and c) Zero-shot NER with TARS (later on TARS) [19].

The Flair Embeddings model uses stacked embeddings, i.e., a combination of contextual string embeddings with a static embeddings model. Contextual string embeddings is a new character based contextual string embeddings method proposed by Akbik et al. [17]. This approach will generate different embeddings for the same word depending on its context. Stacked embedding is an important Flair feature, as a combination of different embeddings might bring better results than their separate uses [7].

The Transformers model or FLERT-extension (document-level features for NER) is a set of settings

---

[2]https://github.com/flairNLP/flair
[3]The release 0.9 (https://github.com/flairNLP/flair/releases/tag/v0.9) was used in the present research. All the descriptions of the Flair framework features refer to the 0.9 release.

[4]AckNER showed better performance as Flair, but is specifically designed to recognize two types of acknowledged entities [4], which was insufficient for the present project.

to perform a NER on document level using fine-tuning and feature-based LSTM-CRF with the multilingual XML-RoBERTa transformer model [18].

The TARS (task-aware representation of sentences) is a transformer-based model, which allows performing training without any training data (zero-shot learning) or with a small dataset (few-short learning) [19]. The TARS approach differs from the traditional transfer learning approach in a way that the TARS model also considers semantic information captured in the class labels themselves. For example, class labels like *food* or *sport* already carry semantic information [19, p. 2].

3. Method

In the present paper, different models for extraction and classification of acknowledged entities were evaluated. The choice of classification was inspired by Giles and Councill [3, p. 17601] classification: funding agencies (FUND), corporations (COR), universities (UNI), and individuals (IND). For our project, this classification was enhanced with the miscellaneous (MISC) and grant numbers (GRNB) categories. The GRNB category was adopted from WoS funding information indexing. The entities in the miscellaneous category could provide useful information but can not be ascribed to other categories, e.g., names of projects and names of conferences. Figure 1 demonstrates an example of acknowledged entities of different types. To the best of our knowledge, Giles and Councill's classification is the only existing classification of acknowledged entities and therefore can be applied for the NER task. Other works on acknowledgement analysis focused on classification of acknowledgement texts.
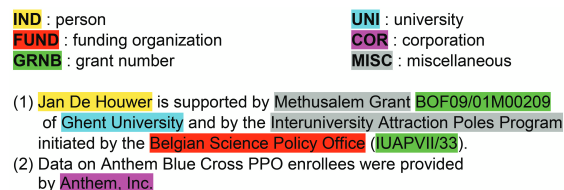
IND : person   **UNI** : university
FUND : funding organization   **COR** : corporation
GRNB : grant number   **MISC** : miscellaneous

(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: An example of acknowledged entities. Each entity type is marked with a distinct color.

Training Data

The Web of Science (WoS) database was used to harvest the training data (funding acknowledgments). From 2008 on, WoS started indexing information about funders and grants. WoS uses information from different fund-

ing reporting systems like researchfish[5], Medline[6] and others. As WoS contains millions of metadata records [20], the data chosen for the present study was restricted by year and scientific domain. 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. Different scientific domains were chosen, as previous works on acknowledgement analysis revealed the relations between scientific domain and types of acknowledged entities, i.e., acknowledged individuals are more characteristic of theoretical- and social-oriented domains, while information about technical and instrumental support are more common for the natural and life science domains [21]. Only the WoS record types *"article"* and *"review"* published in a scientific journal in English were selected; then 1000 distinct acknowledgments texts were randomly gathered from this sample for the training dataset. Further different amount of sentences containing acknowledged entities were distributed into the differently-sized training corpora. Table 1 demonstrates the amount of sentences in each set in the two corpora. We selected only sentences containing an acknowledged entity, regardless of the scientific domain. Table 2 contains the number of sentences from each scientific domain in the training corpora[7].

Corpus No.	Training set (train)	Test set (test)	Validation set (dev)	**Total**
1	29	10	10	**49**
2	339	165	150	**654**

Table 1
Number of sentences in the training corpora.

Corpus No.	Oceanography	Economics	Social Sciences	Computer Science
1	13	3	20	16
2	127	92	351	173

Table 2
Number of **sentences** from each scientific domain in the training corpora.

Preliminary analysis of WoS data showed that WoS funding information indexing has several issues. The WoS includes only acknowledgements containing funding information; therefore, not every database entry has an acknowledgement, individuals are not included, and indexed funding organizations are not divided into different entity types like universities, corporations, etc. Therefore, existing indexing of funding organizations is incomplete. Furthermore, there is a disproportion be-

---

[5]https://mrc.ukri.org/funding/guidance-for-mrc-award-holders/researchfish/
[6]https://www.nlm.nih.gov/bsd/funding_support.html
[7]The same article can belong to several scientific domains.

tween occurrences of acknowledged entities of different types. Thus, the most frequent entity types in the dataset with the training data are IND, FUND and GRNB, followed by UNI and MISC. COR is the most underrepresented category in the dataset. Consequently, there are different amounts of entities of different types in the training corpora (as Figure 2 demonstrates), which might have influenced the training results (see Section 4.1).
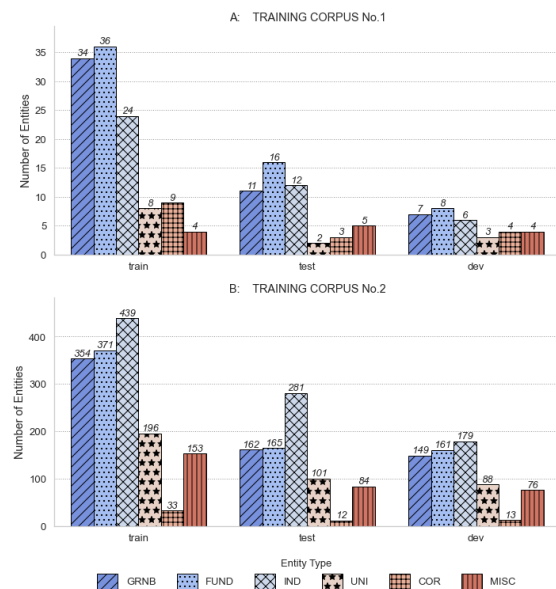

Figure 2: The distribution acknowledged entities of each type in the training corpora.

The training corpus was annotated with six types of entities. As WoS already contains some indexed funding information, it was decided to develop a semi-automated approach for data annotation and use indexed information provided by WoS, therefore, grant numbers were adopted from the WoS indexing unaltered.

Flair has a pre-trained 4-class NER Flair model (CoNLL-03)[8]. The model is able to predict four tags: PER (person name), LOC (location), ORG (organization name), and MISC (other name). As Flair showed adequate results in the extraction of names of individuals, it was decided to apply the pre-trained 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. Furthermore, we noticed that some funding information was partially correctly extracted into the ORG and MISC categories. Therefore, WoS funding organization indexing and entities from the ORG and MISC categories were adopted and distinguished between three categories (FUND, COR and UNI) using regular expres-

---

[8]https://github.com/flairNLP/flair

sions. Further, the automatic classification of entities was manually examined and reviewed. Mismatched categories, partially extracted entities, and not extracted entities were corrected. Acknowledged entities, which fall into the MISC category, were annotated manually.

4. Default Training

Training was performed using three default Flair training algorithms (described in Section 2): Flair Embeddings, Transformers, TARS. The default training was conducted with the recommended parameters for all algorithms, as Flair developers specifically ran various tests to find the best hyperparameters for the default models. Training was conducted with the small (corpus No.1) and bigger (corpus No.2) datasets. The small training corpus was mainly dedicated to test the TARS few-shot scenario. Additionally, we tested a zero-short scenario (without training data) for the TARS model.

Flair Embeddings model uses the combination of static and contextual string embeddings. We applied GloVe [22] as a static word-level embedding model. Thus, in our case stacked embeddings comprise GloVe embeddings, forward contextual string embeddings and backwards contextual string embeddings.

For the Transformers the training was initiated with the RoBERTa model [23]. For the present paper a fine-tuning approach was used. The fine-tuning procedure consisted of adding a linear layer to a transformer and re-training the entire network with a small learning rate. We used a standard approach, where only a linear classifier layer was added on the top of the transformer, as adding the additional CRF decoder between the transformer and linear classifier did not increase accuracy compared with this standard approach [18].

The TARS model requires labels to be defined in a natural language. Therefore we transformed our original coded labels into the natural language: FUND - "Funding Agency", IND - "Person", COR - "Corporation", GRNB - "Grant Number", UNI - "University", and MISC - "Miscellaneous".

4.1. Results

Overall, training demonstrated mixed results. Figure 3-A depicts the training results with the corpus No.1. 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 0.86 (TARS) respectively. The training with Transformers was not successful for IND with the F1-score of 0. Transformers overall showed to be a less efficient algorithm for training with the small dataset with the overall accuracy of 0.35

(Figure 3-C). FUND demonstrated not adequate results with F1-score less than 0.5 for all algorithms. Entity types MISC, UNI and COR showed the worst results with the F1-score equal to zero for all algorithms. Low accuracy for MISC, UNI and COR resulted in low overall accuracy for all algorithms. Overall training with the corpus No.1 showed insufficient results for all algorithms. Flair Embeddings and TARS, though, showed better accuracy in comparison with Transformers. Further, training with the corpus No.2 was performed. Figure 3-B demonstrates training results with the corpus No.2. Similar to the training with corpus No.1, IND and GRNB are the best recognized categories. Best results for IND and GRNB demonstrated Flair Embeddings with the F1-score of 0.98 (IND) and 0.96 (GRNB). TARS achieved the best results for FUND with the F1-score of 0.77 against 0.71 for Flair Embeddings and 0.68 for Transformers. Miscellaneous demonstrated the worst accuracy for Flair Embeddings (0.64) and Transformers (0.49), while for TARS the worst accuracy lies in COR category with the F1-score of 0.54. Best result for UNI showed Flair Embeddings with the F1-score over 0.7.
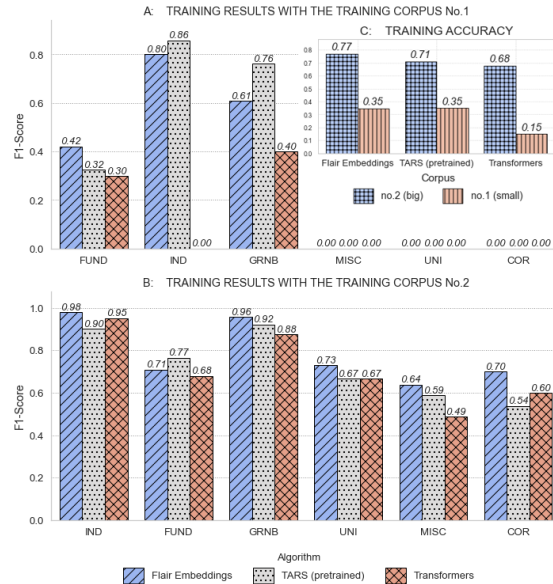

Figure 3: The training results with the training corpora No.1 (Figure A) and No.2 (Figure B). The figure comprises diagrams with the F1-score (for each entity type) of the training with three algorithms. Figure C depicts the total accuracy of training algorithms.

Training with corpus No.2 showed large improvement in training accuracy (Figure 3-C). Overall, Flair Embeddings was more accurate than other training algorithms, although training with TARS showed better results for the FUND category. Transformers showed the worst

results during the training.

Additionally a zero-shot approach was tested for the TARS model on corpus no.1. The model was able to successfully recognize individuals, but struggled with other categories, as Table 3 demonstrates. The total accuracy of the model comprises 0.23.

FUND	GRNB	IND	UNI	COR	MISC
0.23	0.33	0.86	0	0	0

Table 3
F1-Score of each category for the zero-shot learning.

5. Additional training

Our first hypothesis to explain the pure model performance for the FUND, COR, MISC and UNI categories is that they are semantically close, which prevents successful recognition. Entities of these categories are often used in the same context. To examine this hypothesis, we conducted an experiment using Flair Embeddings with the dataset containing three types of entities: IND, GRNB and ORG. The MISC category was excluded from the training, as one of the aims of the present research is to extract information about acknowledged entities and the MISC category contains only additional information. The new ORG category was established, which includes a combination of entities from the FUND, COR and UNI categories. Training was performed with exactly the same parameters as the primary training with the Flair Embeddings model described in Section 4. Results of the training are represented in Figure 4-B. The IND and GRNB still achieved high F1-scores of 0.96 (IND) and 0.95 (GRNB). Nevertheless, ORG gained only a F1-score of 0.64, which is worse than the previous results with six entity types.

The UNI and COR categories, though, have distinct patterns. In this case, the low performance of the models for COR and UNI categories might be explained by the small size of the training sample containing these categories (see Figure 2). Thus, the model was not able to identify patterns because of the lack of data. Secondly, 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 the FUND and COR categories. As a result, training without a MISC category might potentially show better performance. To examine the second hypothesis, 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 4-A. The results were quite similar to those achieved during the training with the dataset with six entity types. Improvement in overall accuracy (Figure 4-D) (0.80 vs. the previous re-

sult of 0.77) could be explained by the fact that the MISC category was not present in this training and could not affect the overall accuracy with its low F1-score.

Additionally, the problem might lie in the nature of training algorithms that were used. On the one hand, Flair developers claimed Transformers to be the most efficient algorithm [18]. On the other, the stacked embeddings are an important feature of the Flair tool, as a combination of different embeddings might bring better results than their separate uses [7]. Thus, the combination of the Transformer embeddings model with the contextual string embeddings might improve the model performance. Thus, for the third additional training we combined contextual string embeddings with FLERT parameters. The training results are represented in Figure 4-C. The proposed method showed no improvements compared to the results of the primary training with Transformers and worse performance compared with Flair Embeddings.
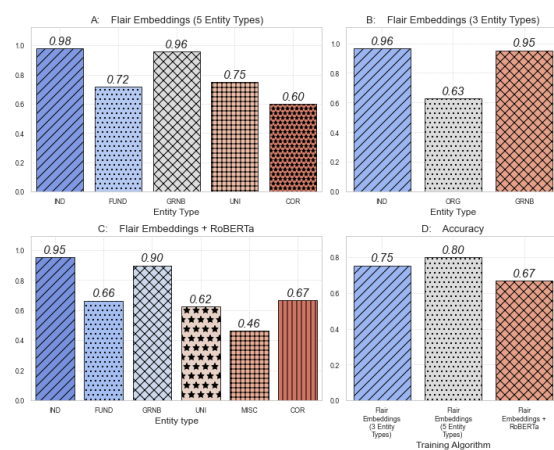

Figure 4: The results of the additional training. Figures A, B, C comprise diagrams with the F1-score of the additional training with three algorithms. Figure D represents the total accuracy of the training algorithms.

Discussion

The results of the additional training generally showed no improvement in the accuracy. On the contrary, training with the three entity types deteriorated the model performance. That might indicate that the model would make better predictions if the number of entity types is expanded. For example, the MISC category could be split into the following categories: names of projects, names of conferences, names of software and miscellaneous (for other possible information). Different subcategories could also be distinguished in the FUND category. Thus, this hypothesis requires further investigation.

The model performance with Transformers and Flair Embeddings could also be improved by expanding the training corpus and adding more entries containing entities with the low recognition accuracy. Moreover using another transformer model such as SciBert [24] might increase the model performance.

6. Conclusion

In this paper we evaluated different embedding models for the task of automatic extraction and classification of acknowledged entities from acknowledgement texts[9]. The annotation of the training corpora was the most challenging and time-consuming task of all data preparation procedures. Therefore, a semi-automated approach was used to help significantly accelerate the procedure. Flair Embeddings showed the best accuracy in the training with the bigger corpus and the fastest training time in comparison with the other models; thus, it is recommended to further use the Flair Embeddings model for the recognition of acknowledged entities. Expanding the size of a training corpus massively increased the accuracy of all training algorithms. Main limitations of the study were the small sizes and just one annotator of the training corpora.

References

[1] J. P. Kassirer, M. Angell, On authorship and acknowledgments, The New England journal of medicine 325 (1991) 1510–1512. doi:10.1056/NEJM199111213252112.

[2] B. Cronin, S. Weaver, The praxis of acknowledgement: from bibliometrics to influmetrics, Revista Española de Documentación Científica 18 (1995) 172. Place: Madrid Publisher: CENIDOC.

[3] C. L. Giles, I. G. Councill, Who gets acknowledged: Measuring scientific contributions through

---

[9]The best model can be tested via an online demo available at: https://colab.research.google.com/drive/1Wz4ae5c65VDWanY3Vo-fj_bFjn-loL4?usp=sharing

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

[11]https://www.bibliometrie.info/en/index.php?id=home

automatic acknowledgment indexing, Proceedings of the National Academy of Sciences 101 (2004) 17599–17604. URL: http://www.pnas.org/cgi/doi/10.1073/pnas.0407743101. doi:10.1073/pnas.0407743101.

[4] D. Alexandera, A. P. Vries, "This research is funded by...": Named Entity Recognition of financial information in research papers, BIR 2021: 11th International Workshop on Bibliometric-enhanced Information Retrieval at ECIR (2021) 102–110. URL: http://ceur-ws.org/Vol-2847/paper-10.pdf.

[5] S. Kayal, Z. Afzal, G. Tsatsaronis, S. Katrenko, P. Coupet, M. Doornenbal, M. Gregory, Tagging funding agencies and grants in scientific articles using sequential learning models, in: BioNLP 2017, Association for Computational Linguistics, Vancouver, Canada,, 2017, pp. 216–221. URL: https://aclanthology.org/W17-2327. doi:10.18653/v1/W17-2327.

[6] M. Song, K. Y. Kang, T. Timakum, X. Zhang, Examining influential factors for acknowledgements classification using supervised learning, PLOS ONE (2020). URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0228928.

[7] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, R. Vollgraf, FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 54–59. doi:10.18653/v1/N19-4010.

[8] A. Paul-Hus, A. Díaz-Faes, M. Sainte-Marie, N. Desrochers, R. Costas, V. Larivière, Beyond funding: Acknowledgement patterns in biomedical, natural and social sciences, PLOS ONE 12 (2017) e0185578. doi:10.1371/journal.pone.0185578.

[9] A. Paul-Hus, N. Desrochers, Acknowledgements are not just thank you notes: A qualitative analysis of acknowledgements content in scientific articles and reviews published in 2015, PLOS ONE 14 (2019) e0226727. doi:10.1371/journal.pone.0226727.

[10] K. Mccain, Beyond garfield's citation index: an assessment of some issues in building a personal name acknowledgments index, Scientometrics 114 (2017). doi:10.1007/s11192-017-2598-1.

[11] A. K. Thomer, N. M. Weber, Using named entity recognition as a classification heuristic, in: iConference 2014 Proceedings, iSchools, 2014, pp. 1133 – 1138. URL: http://hdl.handle.net/2142/47362. doi:https://doi.org/10.9776/14401.

[12] DeepAI, Named-entity recognition, 2019. URL: https://deepai.org/machine-learning-glossary-and-terms/named-entity-recognition.

[13] K. Hub, What is Semantic Annotation?, ???? URL: https://www.ontotext.com/knowledgehub/fundamentals/semantic-annotation/.

[14] J. Pustejovsky, A. Stubbs, Natural Language Annotation for Machine Learning, O'Reilly Media, Inc., 2012. URL: https://learning.oreilly.com/library/view/natural-language-annotation/9781449332693/.

[15] J. R. Finkel, T. Grenager, C. Manning, Incorporating non-local information into information extraction systems by Gibbs sampling, in: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05), Association for Computational Linguistics, Ann Arbor, Michigan, 2005, pp. 363–370. URL: https://aclanthology.org/P05-1045. doi:10.3115/1219840.1219885.

[16] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: An Imperative Style, High-Performance Deep Learning Library, arXiv:1912.01703 [cs, stat] (2019). URL: http://arxiv.org/abs/1912.01703, arXiv: 1912.01703.

[17] A. Akbik, D. Blythe, R. Vollgraf, Contextual String Embeddings for Sequence Labeling, 2018, 27th International Conference on Computational Linguistics (2018) 1638–1649.

[18] S. Schweter, A. Akbik, FLERT: Document-Level Features for Named Entity Recognition, ArXiv (2020).

[19] K. Halder, A. Akbik, J. Krapac, R. Vollgraf, Task-Aware Representation of Sentences for Generic Text Classification, in: Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 3202–3213. URL: https://aclanthology.org/2020.coling-main.285. doi:10.18653/v1/2020.coling-main.285.

[20] V. K. Singh, P. Singh, M. Karmakar, J. Leta, P. Mayr, The Journal Coverage of Web of Science, Scopus and Dimensions: A Comparative Analysis, Scientometrics 126 (2021) 5113–5142. URL: http://arxiv.org/abs/2011.00223. doi:10.1007/s11192-021-03948-5.

[21] A. A. Diaz-Faes, M. Bordons, Making visible the invisible through the analysis of acknowledgements in the humanities, Aslib Journal of Information Management 69 (2017) 576–590. URL: https://arxiv.org/pdf/1706.06334. doi:10.1108/AJIM-01-2017-0008.

[22] J. Pennington, R. Socher, C. D. Manning, GloVe: Global Vectors for Word Representation, in: Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1532–1543. URL: http://www.

aclweb.org/anthology/D14-1162.

[23] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A Robustly Optimized BERT Pretraining Approach, arXiv:1907.11692 [cs] (2019). URL: http://arxiv.org/abs/1907.11692, arXiv: 1907.11692.

[24] I. Beltagy, K. Lo, A. Cohan, SciBERT: A pretrained language model for scientific text, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3615–3620. URL: https://aclanthology.org/D19-1371. doi:10.18653/v1/D19-1371.