The Automatic Detection of Dataset Names in Scientific Articles

Heddes, J.; Meerdink, P.; Pieters, M.; Marx, M.

DOI
10.3390/data6080084
Publication date
2021
Document Version
Final published version
Published in
Data
License
CC BY

data

MDPI

Data Descriptor

The Automatic Detection of Dataset Names in Scientific Articles

Jenny Heddes, Pim Meerdink, Miguel Pieters and Maarten Marx *

Informatics Institute, Faculty of Science, University of Amsterdam, Science Park 908, 1098 XH Amsterdam, The Netherlands; jenny.heddes@student.uva.nl (J.H.); pim.meerdink@student.uva.nl (P.M.); miguel.pieters@student.uva.nl (M.P.)
* Correspondence: maartenmarx@uva.nl

Abstract: We study the task of recognizing named datasets in scientific articles as a Named Entity Recognition (NER) problem. Noticing that available annotated datasets were not adequate for our goals, we annotated 6000 sentences extracted from four major AI conferences, with roughly half of them containing one or more named datasets. A distinguishing feature of this set is the many sentences using enumerations, conjunctions and ellipses, resulting in long BI+ tag sequences. On all measures, the SciBERT NER tagger performed best and most robustly. Our baseline rule based tagger performed remarkably well and better than several state-of-the-art methods. The gold standard dataset, with links and offsets from each sentence to the (open access available) articles together with the annotation guidelines and all code used in the experiments, is available on GitHub.

Dataset: https://github.com/xjaeh/ner_dataset_recognition

Keywords: dataset extraction; scientific information extraction; named entity recognition; BERT; SciBERT

Citation: Heddes, J.; Meerdink, P.; Pieters, M.; Marx, M. The Automatic Detection of Dataset Names in Scientific Articles. *Data* **2021**, *6*, 84. https://doi.org/10.3390/data6080084

Academic Editor: Craig A. Knoblock

Received: 16 June 2021
Accepted: 28 July 2021
Published: 4 August 2021

1. Introduction

This paper contributes to the creation of a *dataset citation network*, a knowledge graph linking datasets to scientific articles when used in an article. Unlike the citation network of papers, the dataset citation infrastructure is still primitive, due to the limited referencing of dataset usage in scientific articles [1–4]. The use and value of such a dataset citation network is similar to that of the ordinary scientific citation network: realizing recognition for dataset providers by computing the impact scores of datasets based on citations [2,4], ranking datasets in dataset search engines by impact [1], creating a representation of a dataset by its use instead of its metadata and content [4,5], studying cooccurrences of datasets, etc. According to Kratz and Strasser [2], researchers believe that the citation count is the most valuable way to measure the impact of a dataset.

Creating the dataset citation network from a collection of articles involves three main steps: scientific PDF parsing, recognizing and extracting mentioned datasets, and cross documenting the coreference resolution ("dataset name de-duplication"). This paper is only concerned with the dataset extraction process, which we view as a Named Entity Recognition (NER) task. When focusing on the articles that use a NER method for this task, it becomes clear that almost every article uses another approach and another dataset. Not only do these approaches differ, but they also deviate from the core dataset NER task, as every approach has something extra added onto it [3,6–14]. This makes it hard to compare which method or component fits a task best. According to Beltagy et al. [15], SciBERT has shown state-of-the-art results on one of the datasets (SciERC), while other methods have outperformed this score on a similar task and dataset [9,10]. To fully be able to compare the performance and annotation costs of each (basic) model, we compare their

performance with them all being trained and tested on the same dataset. This results in the following research question:

RQ Which Named Entity Recognition model is best suited for the dataset name recognition task in scientific articles, considering both performance and annotation costs?

The comparison of the performance of each method, when only run once, is not sufficient enough to fully compare them, as [16] showed that annotation choices have an irrefutable impact on the system's performance. This effect was neglected in the aforementioned papers that focused on the dataset extraction task. However, not only can these choices impact the models' performances, but they can also impact the annotation costs. We consider a number of factors that could influence the performance and annotation costs of the models. First, domain transfer is considered, as this is shown to impact NER performance [17–19]. This has become a trending topic in NER in an effort to reduce the amount of training data needed [20]. Another factor that is taken into account is the training set size, as multiple sources have shown that a small amount of training data can lead to performance problems [7,21]. Next to the size of the training set, the effect of the distribution of positive and negative samples is considered, as this, too, has been shown to influence performance [22–25]. These choices all influence the amount of training data that is needed to achieve the best performance, thus influencing the annotation costs since adding 'real examples' is costly. In order to further reduce the annotation costs, the effect of adding weakly supervised examples in the training data is investigated for the best performing model. Summing up, we answer the following questions:

RQ1 What is the performance of rule-based, CRF, BiLSTM, BiLSTM-CRF [26–28], BERT [29] and SciBERT [15] models on the dataset name recognition task?

RQ2 How well do the models perform when tested on a scientific (sub)domain that is not included in the training data?

RQ3 How does the amount of training data impact the models' performance?

RQ4 How are the models' performance affected by the ratio of negative versus positive examples in the training and test data?

RQ5 Does adding weakly supervised examples further improve the scores of the best performing model? Additionally, how well does the best performing model perform without any manually annotated labels?

RQ6 Is there a difference in the performance of NER models when predicting easy (short) or hard (long) dataset mentions?

To answer these questions on realistic input data, we created a hand-annotated dataset of 6000 sentences based on four sets of conferences in the fields of neural machine learning, data mining, information retrieval and computer vision (see Section 3.1). NER can be evaluated in many ways. We mostly use the most strict and realistic, that is, the exact match on a zero shot test set. We note, however, that, due to enumerations and ellipses, many NER hits contain several datasets, which makes the partial and B-match also useful (as the found NER hits have to be post-processed anyway).

Our main findings are that SciBERT performs best, particularly on realistic test data (with >90% sentences, not mentioning a dataset). Surprisingly our own developed rule based system (using POS tags and keywords) performed almost as well, and all others, except BERT, perform (much) worse than this rule-based system. SciBERT was also robust when looking at the other tests performed, regarding domain adaptability, the negative sample ratio and the train set size. However, nothing comes for free; we did not succeed in training SciBERT to outperform the rule-based system when we gave it only weakly supervised training examples (obtained without manual annotation).

All code and datasets used for this paper can be found at https://github.com/xjaeh/ner_dataset_recognition.

2. Related Work

The overwhelming volume of scientific papers have made extracting knowledge from them an unmanageable task [30], making automatic IE especially relevant for this domain [31]. Scientific IE has been of interest since the early 1990s [32]. Despite the growing interest in the automatic extraction of scientific information, research on this topic is still narrow even now [7]. The reason for the limited research in scientific IE in comparison to the general domain is the specific set of challenges associated with the scientific domain. The main challenge is the expertise that is needed for annotated data, making these data costly and hard to obtain, resulting in very limited data available [7]. However, there is a significant focus on this kind of research in the scientific sub-domains: medicine and biology [30].

Where at the beginning of Scientific IE, the focus mainly laid upon citations and topic analyses [13], now, the focus has become broader and has shifted toward scientific fact extraction (for example, population statistics, variants of genomics, material properties, etc.) [30]. Research on the dataset name extraction task uses a great variety of methods throughout the NER spectrum, including, but not limited to, the following: rule-based, BiLSTM-CRF and BERT [3,6–14].

For dataset extraction, it was found that verbs surrounding the dataset provide information about the role or function; as such, the words, use, apply or adopt, indicate a 'use' function [10]. Nevertheless, not only these verbs surrounding it play an important role, as for dataset detection, a wide range of context is needed [6], indicating that a model's ability to grasp context could play a significant role in the performance of that model.

We briefly go through the NER models that we tested for dataset extraction.

The rule-based approach was the most prominent one in the early stages of NER [33]. Despite the fact that most state-of-the-art results are now achieved by machine learning methods, the rule-based model is still attractive to use, due to its transparency [34]. The authors conclude that rule-based methods can achieve state-of-the-art extraction performance, but note that the rule development is very time consuming and a manual task. Not only is this method used as a stand-alone classification method, but it is also suitable as a form of weak supervision [35] as an alternative to the manual labeling of data, providing training examples for the other methods [36,37].

Conditional Random Fields (CRF) is a probabilistic model for labeling sequential data, which has proven its effectiveness in NER, producing state-of-the-art results around the year 2007 [38]. A dataset extraction model solely based on CRF is missing, but it was used for other tasks in Scientific IE. A well-known example is the GROBID parser, which extracts bibliographic data from scientific texts (such as the title, headers, references, etc.) [39].

The BiLSTM-CRF is a hybrid model, combining LSTM layers with a CRF layer on top [40]. Using this combination, the advantages of both models can be joined. The advantage of BiLSTM is that it is better at predicting long sequences, predicting every word individually [41], while CRF predicts based on the joint probability of the whole sentence, making sure that the optimal sequence of tags is achieved [41–43]. To date, the BiLSTM-CRF based model produces the best performance on the dataset extraction task, with an F1 score of 0.85 [8].

BERT produces state-of-the-art results in a range of NLP tasks [29]. It is based on a transformer network, which is praised for its context-aware word representations, improving the prediction ability [44]. BERT has revolutionized classical NLP. However, its performance as a base for the dataset extraction tasks differs greatly, as one research study found an F1 score of 0.68 [13], while another has found an F1 score of 0.79 [10]. Beltagy et al. [15] developed the SciBERT model based on the BERT model. The big and only difference between those models is that, unlike BERT, which is trained on general texts, SciBERT is trained on 1.14 M scientific papers from Semantic Scholar, consisting of 18% computer science papers, and the remaining 82% consisting of papers from the biomedical domain. This model, which was specially created for knowledge extraction in the scientific

domain, indeed achieves better performance, in comparison to BERT, in the computer science domain.

3. Materials and Methods

3.1. Description of the Data

We describe the created manually annotated dataset. The annotation guidelines in Appendix B contain many illuminating examples. Here, we simply give two examples (annotated datasets are in marked in gray):

- *The second collection (called ClueWeb) that we used is ClueWeb09 Category B , a large-scale web collection . . .*
- *Tables 3 and 4 show the average precision of 20 categories and MAP on the PASCAL VOC 2007 and 2012 testing set , respectively.*

3.1.1. Origins

The sentences in the dataset originate from published articles from four different corpora within the computer science domain: the Conference on Neural Information Processing Systems (NIPS, 2000–2019), SIAM International Conference on Data Mining (SDM, 2000–2019), ACM SIGIR conference (2007–2018), and papers from the main conferences (ICCC, CVPR) participating in the Computer Vision foundation (VISION, 2017–2019). These conferences were chosen because they are top tier $A*$ venues that have existed for an extensive period; all of them, except SIGIR, are freely available; they cover a wide range of topics within the information sciences; and experimental evaluation is a key aspect in these venues. Thus, we expected most articles to contain references to the datasets. Papers from these conferences were collected in PDF format and parsed using GROBID to extract the text [39]. The extraction of sentences using GROBID made it possible to exclude references, titles and tables. This way, only 'real' sentences from the main text were selected. From these sentences occurring in the main text, we selected sentences that likely contained a reference to a dataset for manual annotation. These selected sentences had to contain one of the following phrases (including their plural form): dataset, data set, database, data base, corpus, treebank, benchmark, test/validation/testing/training data or train/test/validation/testing/training set. The regular expression in Appendix A was used to implement this selection.

3.1.2. Annotation

The annotation scheme used for the annotation task is based on the ACL RD-TEC Annotation Guidelines [45]. An example of a guideline is that generic nouns (e.g., dataset) accompanying a term should be annotated. Another example is the 'ellipses rule', which states that when two noun phrases in a conjunction are linked through ellipses, the term needs to be annotated as one. For the task of the dataset name annotation, this would mean that the phrase *PASCAL VOC 2006 and 2007 datasets* are marked as one entity. Annotation was done by four persons, each annotating 1500 sentences plus a part of the kappa calculation. The resulting Fleiss kappa of 0.72 representing a substantial agreement was calculated based on fifty sentences [46]. The full annotation scheme is available in Appendix B.

3.1.3. Train and Test Sets

Each annotated sentence was given an ID, tokenized using the spaCy tokenizer [47], and given POS-tags using the NLTK package [48]. The gold standard annotations themselves were transformed into the corresponding IOB-tags for each token.

The entire dataset contains a total of 6000 sentences, having an even distribution between corpora, with 1500 sentences from each corpus. Slightly under half of them contain a dataset mention. These 6000 sentences were split into a train set, test set and zero shot test set. Sentences in the zero shot test set do not contain dataset names that occur in the train set. All sets were created using stratified sampling, more or less keeping the equal

distribution among the four conferences. The distribution of positive and negative samples in these sub-sets is shown in Table 1.

Table 1. Distribution of positive and negative samples across the train, test and zero-shot set.

	Containing a Dataset	No Dataset	Total
Training set	2168	2472	4640
Test set	543	617	1160
Zero-shot set	200	0	200
Total	2911	3089	6000

The number of sentences containing a dataset name is not equal to the number of datasets being named, which is 4164. This leaves an average mention of 1.43 datasets in a sentence containing at least one dataset mention and an average mention of 0.69 overall in this dataset.

We ran the SciBERT tagger over the complete corpus of over 15,000 articles, and observed that within the VISION papers, 5% of the papers did not mention a dataset. For the other three conferences, this was remarkably similar between 20.2 and 22.4%. A manual total scan of 30 random NIPS papers produced a slightly higher part of nine papers without any dataset mention.

3.2. Experimental Setup

Full details of all experiments, including more detailed measurements, are available on the GitHub repository. All methods were evaluated using seqeval [49] for the B- and I-tags, and nervaluate [50] for the partial- and exact-match scores.

As a natural baseline, we created a rule-based system containing rules such as "*If the word is a proper-noun and one of the keywords follows: then mark the proper-noun including its keyword as a dataset*", which were developed through careful consideration, following the annotation guidelines. As developing a rule-based system takes time [34], the rule development was an iterative process by trial and error, each time adding or adjusting rules as deemed necessary. The rules were made machine readable, using spaCy's rule-based matching method [51]. This translated to 10 spaCy *patterns*; see the notebook Rule-based.ipynb in the dataset belonging to this paper.

For the CRF, the sklearn_crfsuite from the scikit-learn library was used [52]. Both BiLSTM methods are keras based. No parameter optimization was performed. The used parameters were taken from the "Depends on the definition NER series" [53].

Both BERT models are based upon a scikit-learn wrapper [54]. This wrapper provides multiple models that can be selected. The example of [29] is followed by choosing the cased model for NER. While the uncased variant of the model generally performs better, the choice was made to utilize a case-sensitive model, as capitalization can be indicative of whether a phrase refers to a dataset: words referring to datasets are often capitalized. To compare both models equally, the $\text{BERT}_{\text{base}}$ model was chosen, just like [15], as SciBERT only has a base model. For SciBERT, the scivocab was chosen, as this represents the frequently used words in scientific papers. The model configuration and architecture are the same as those in the SciBERT paper [15]. The following hyperparameters were used for the training of the model: A learning rate of 5×10^{-6} for the Adam optimizer, with a batch size of 16. Training lasted 15 epochs, and checkpoints were saved every 300 training steps. Gradient clipping was used, with a max gradient of 1.

4. Results

We report the results grouped by the five subquestions. Appendix D contains additional results (e.g., precision and recall scores, and scores for the B(eginning) and I(nternal) tags.

4.1. RQ1, Overall Performances

Table 2 contains the F1 performance scores for the six different NER models we tested. This is the only experiment we conducted with (5-fold) cross validation on the complete set of 6000 sentences. BERT and SciBERT perform almost the same on both scores and (much) better than all the others, except that the rule-based system performs equally well on the partial match score.

Notice that the partial and exact match scores are closest for SciBERT. Due to conjunctions, ellipses and the used annotation guidelines, NER phrases can be quite long, so a large difference between the two ways of scoring could be expected. An error analysis shows that SciBERT is especially good in learning the *beginning* of a dataset mention.

The two most interesting systems seem to be SciBERT and the rule-based one, and thus we will mostly report results on the other subquestions for these two.

Table 2. RQ1, Partial and exact match mean F1 scores for various NER models based on 5-fold cross-validation, using the complete dataset of 6000 sentences (all standard deviations are between 0.01 and 0.03).

	Partial Match	**Exact Match**
Rule Based	0.81	0.72
CRF	0.75	0.72
BiLSTM	0.73	0.67
BiLSTM-CRF	0.77	0.72
BERT	0.81	0.77
SciBERT	0.82	0.78

4.2. RQ2, Domain Adaptability

The models' ability to adapt to differences within the scientific domain is shown in Table 3. These scores are achieved using one corpus as a test set, while training on the other three corpora. The corpus on which it is tested can be found in the header. We expected the scores to be lower than the cross validation scores, but we only found a small negative effect when testing on the VISION conferences. We note that the VISION set is different in that the sentences come from the last three years, while the others are from the last two decades.

Table 3. RQ2, domain adaptability between corpora (F1 scores).

Models	**Evaluation**	**NIPS**	**SDM**	**SIGIR**	**VISION**
Rule-based	Partial-match	0.75	0.75	0.79	0.75
	Exact-match	0.72	0.70	0.75	0.71
SciBERT	Partial-match	0.81	0.83	0.80	0.78
	Exact-match	0.76	0.80	0.76	0.71

4.3. RQ3, Amount of Training Data

Here, we look at the major cost factor: the size of the training set. Figure 1 shows the exact match F1 score on the zero-shot set for varying amounts of training sentences, ranging from 500 to 4500. We see a clear difference between CRF and the two BERT models on the one hand and the two BiLSTM models on the other. We now zoom in on the most stable behaving models, CRF and SciBERT. Figure 2 zooms in on both precision and recall, also for the (supposedly easier) test set. Both models show remarkably robust behavior: only a slight influence of the amount of training examples and hardly any difference in performance for the test and zero-shot test set. It is noticeable that CRF can be seen as a precision-oriented system, while for SciBERT, precision and recall are very similar. We see this as evidence that these two systems learn the structure of a dataset mention well and do not overfit on the dataset names themselves.

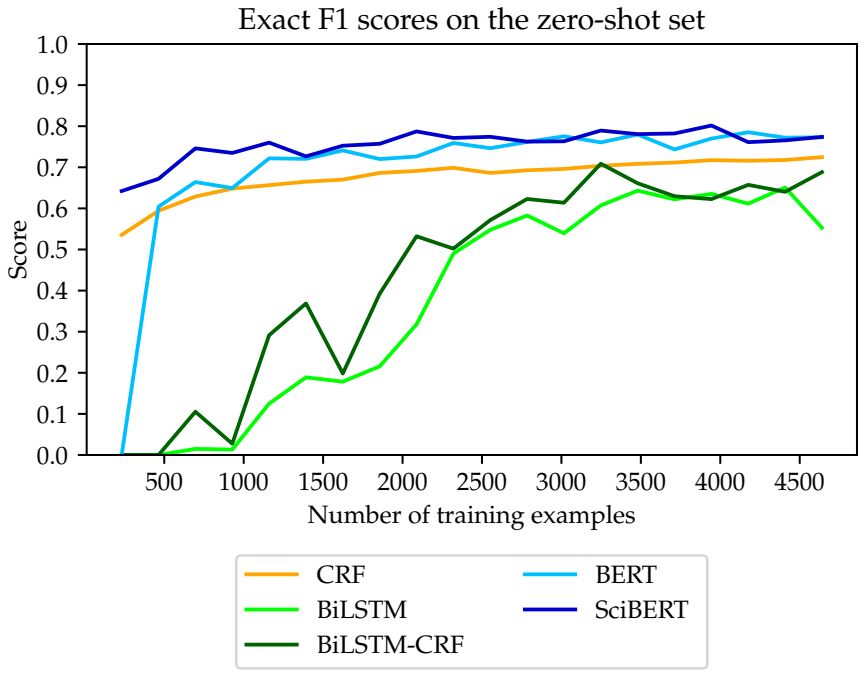

Figure 1. RQ3, the influence of the amount of training data for all models tested on the zero-shot test set.

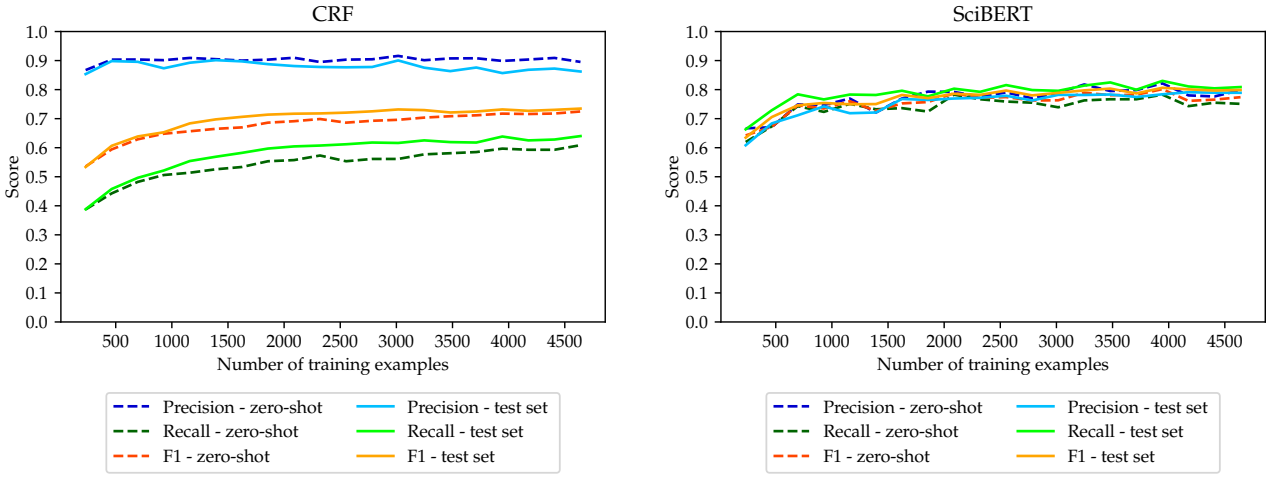

Figure 2. RQ3, the influence of the amount of training data for the CRF model (**left**) and for SciBERT (**right**).

4.4. RQ4, Negative/Positive Ratio

Recall that about half of the sentences in our dataset do not mention a dataset, *while containing one of the trigger words, such as dataset, corpus, collection, etc*. We can decide to use those in training or not. As noted in previous research, the ratio of positive and negative sentences was found to be important for NER models trained on a dataset mention extraction task [6]. We see a slight improvement in F1 scores for all models when adding also negative training examples, but this is quite small.

What is more interesting is when we test on a set in which sentences mentioning a dataset are very rare, just like in a real scientific article. Using the developed rule-based system, we added sentences that most probably do not mention a dataset (i.e., they did not contain any of the trigger words (more precisely, did not match the regex in Appendix A)) to the test set until we obtained a 1 in 100 ratio. Table 4 shows the results. We see that all F1 scores drop, compared to those in Table 2. This is expected as the task becomes harder.

However, note that the recall remains very high for the two BERT models, indicating that the drop in F1 is caused mainly by extra false positives (of course, it might be that the SciBERT model discovered genuine dataset mentions not containing one of the trigger terms. We did not check for this).

Table 4. RQ4, testing on a real ratio (positive vs. negative) test set.

Model	Evaluation	Precision	Recall	F1
CRF	Partial-match	0.72	0.66	0.69
	Exact-match	0.69	0.63	0.66
BiLSTM	Partial-match	0.37	0.19	0.26
	Exact-match	0.29	0.15	0.20
BiLSTM-CRF	Partial-match	0.49	0.29	0.37
	Exact-match	0.38	0.23	0.29
BERT	Partial-match	0.57	0.91	0.70
	Exact-match	0.55	0.88	0.68
SciBERT	Partial-match	0.65	0.91	0.76
	Exact-match	0.63	0.88	0.73

4.5. RQ5, Weakly Annotated Data

We now see how much SciBERT can learn from positive training examples discovered by the rule-based system. As these examples are not hand annotated, we call them weakly supervised. We created a weakly supervised training set, SSC (for Silver Standard Corpus), with the same number of positive and negative sentences as in the manually annotated train set. Table 5 shows that the performance of SciBERT is substantially lower when trained on those 'cost-free' training examples alone than when trained on the hand-annotated data (train set). A reason for this is that the SSC can contain false negatives or false positives, and learning from these false data will impact the model's prediction ability, thus impacting the scores.

Table 5. RQ5, differences between training on supervised or weakly supervised train data (F1 scores).

Training Set	Evaluation	Test Set	Zero-Shot
Train set	Partial	0.83	0.81
	Exact	0.80	0.76
SSC	Partial	0.73	0.73
	Exact	0.67	0.67

According to [55], weakly supervised negative examples harm the performance. To test this effect, SciBERT was also trained on a combination of the manually labeled data and only the positive data from the SSC. The differences were very small: a 0.01 improvement for both partial and exact match on the zero-shot test, no difference for the partial match, and a 0.02 decrease on the test set.

4.6. RQ6, Easy vs. Hard Sentences

Sentences enumerating a number of named datasets are common in scientific articles. According to the guidelines, these are tagged as one entity, leading to long BI+ tag sequences. We wanted to test whether SciBERT is able to learn these more complex long entities just as well as the easier ones. So we split both the train and the zero-shot test sets into a hard and easy set, with sentences being hard if they contained a BI+ tag sequence of a length of four or more. We then performed all four possible train on hard/easy, test on hard/easy experiments. Only (we also saw a 6% drop in F1 when trained on hard and

test on easy, but this may be due to much less training sentences) with train on easy, test on hard did we see an expected but still remarkable difference in scores (a drop in F1 of 42%). This means that the network is also able to understand and interpret ellipses and enumerations. These more complex rules and structures are not harder for the network to identify than simple one- or two-word dataset mentions. These structures and patterns are difficult, even for human annotators to consistently parse and classify correctly, making the network's ability to understand the nuances of the labeling task significant.

5. Discussion

We have created a large and varied annotated set of sentences likely to contain a dataset name, with about half actually containing one or more datasets. We have shown that extracting these datasets using traditional NER techniques is feasible but clearly not straightforward or solved. We believe our results show that the created gold standard is a valuable asset for the scientific document parsing community. The set stands out because the sentences come from all sections in scientific articles, and come with exact links to the articles. Except for those coming from SIGIR, all articles are openly available in PDF format.

Analysis of the errors of the NER systems and the disagreements among the annotators revealed that dataset entity recognition from scientific articles is complicated through the use of enumerations, conjunctions and ellipsis in sentences. This means that, for example, in the sentence '*We used the VOC 2007, 2008 and 2009 collections.*', the phrase 'VOC 2007, 2008 and 2009 collections' is tagged as one dataset entity mention, as individual elements of the enumeration are nonsensical without the context provided by the other elements [7]. We think it is this aspect that makes the task exciting and different from standard NER. Postprocessing the found mention, extracting all dataset entities, and completing the information hidden by the use of ellipses is an NLP task needed on top of dataset NER before we can create a dataset citation network. Of course, a cross-document coreference resolution of the found dataset names is then needed for the obtained network to be useful [56]. Expanding the provided set of sentences with this extra information, linking every sentence to a set of unique dataset identifiers is not that much work and would make the dataset also applicable for training the dataset reconciliation task.

We wanted to know which NER system performs well and at what cost. Not surprisingly, the best performing systems were BERT and SciBERT. Unsupervised pretraining also helps for this task. Both systems (and CRF) worked already almost optimally with relatively few training examples. They were robust on our domain adaptation experiments, and kept a high recall at the cost of some loss in precision when we diluted the test set to a realistic 1 in 100 ratio of sentences with a dataset.

We found the performance of our quite simple rule-based system to be remarkable. In fact, this system can be seen as a formalization of the annotation guidelines, and having those carefully spelled out made it almost effortless to create; this is, in our opinion, the reason for its strong performance. The experiment in which we trained SciBERT with extra examples found by the rule-based system was inconclusive in that we saw hardly any change in performance. However, there may be more clever ways to combine these two models.

Future Directions

We think a gold standard dataset reminiscent of the end-to-end task of dataset mention extraction from scientific PDFs could lead to a big step forward in this field. In particular, we could then train and test end-to-end systems, which would link dataset DOIs to article DOIs.

Additionally, the articles from the four chosen ML/DM/IR/CV conferences are relatively easy for the dataset extraction task, as they do not contain that many named entities. The task is likely harder with papers from biological, chemical or medical domains.

A different approach to this task is to start with a knowledge base of existing research datasets containing their names and some metadata and then to use that in an informed dataset mention extraction system.

Author Contributions: Conceptualization, J.H. and M.M.; methodology, J.H., P.M., M.M.; software, J.H., P.M., M.P., M.M.; validation, J.H., P.M., M.P.; formal analysis, J.H., P.M., M.M.; investigation, J.H., P.M., M.M.; resources, J.H., P.M., M.P.; data curation, J.H., P.M., M.P.; writing—original draft preparation, J.H.; writing—review and editing, M.M.; visualization, J.H.; supervision, M.M.; project administration, M.M.; funding acquisition, M.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the Netherlands Organization for Scientific Research (NWO, ACCESS project, grant No. CISC.CC.016).

Data Availability Statement: The gold standard dataset, with links and offsets from each sentence to the (open access available) articles together with the annotation guidelines and all code used in the experiments, is available on https://github.com/xjaeh/ner_dataset_recognition.

Acknowledgments: Special thanks to Wouter Haak and Elsevier for the valuable comments and providing additional datasets.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Appendix A. REGEX

The used regular expression was `(D)` with `D` a isjunction of the following expressions: `(?:train|test|validation|testing|trainings?)\s*(?:set|data), benchmarks?, data\s*(?:set|base)s?, corpus, corpora, tree\s*bank, collections?.`

Appendix B. Annotation Guidelines

Appendix B.1. Introduction

The main idea of this task is to mark the dataset name(s) in sentences. The Oxford dictionary defines name as follows: "a word or set of words by which a person or thing is known, addressed, or referred to" [57]. A dataset is defined as a collection of data that is treated as a single unit by a computer [58]; so mark the word or a set of words that refers to a dataset. This leads to the following definition.

Dataset name: Term(s) that refers to a real (life) dataset.

The idea is to find the dataset name(s) in sentences and to mark them so that they contain the needed information to find the dataset online (so that versions are included). The sample sentences below are sentences from the NIPS papers [59].

The following are a few examples of sentences without a dataset name:

- *First, let us remark that our training setup differs from those reported in previous works.*
- *Section 3 demonstrates the validity of our approach by evaluating and comparing the model on a number of image datasets.*
- *This dataset consists of about 5000 trainval images and 5000 test images over 20 object categories.*

The following are a few annotated examples of sentences containing dataset names.

- *Over two benchmark datasets, Standford Background and Sift Flow, the model outperforms many state-of-the-art models in accuracy and efficiency.*
- *Imagenet : A large-scale hierarchical image database.*
- *For one, we trained on the standard WSJ training dataset .*
- *In all experiments, we used stack1 for testing, stack2 and stack3 for training, and stack4 as additional training data for recursive training.*

- *We tested our formulation on three datasets of object detection:* *PASCAL VOC 2007*, *PASCAL 2012* *and* *Microsoft COCO*.

Appendix B.2. Guidelines for Annotation

- Mark entire word (an—does not split a word).
- Mark a dataset name consisting of multiple words as one; when marked as individual words, they will be seen as multiple names.
- Do not mark the reading signs/spaces before or after the name.

Appendix B.3. When to Annotate

- Datasets can be referred to in a number of different ways, including, but not limited to the following: database, benchmark, collection, corpus, treebank, etc.
- Only mark a name when it is specific enough.
  - This does not include the following:
    * When talking about 'a' dataset.
      · *Our model performs 1.8 times better than the only published results on an existing image QA dataset.*
    * When a dataset name is mentioned, but the data set is not talked about.
      · *It should be noted that our DAQUAR results are for the portion of the dataset (98.3%) with single-word answers.*
    * When they only mention data.
      · *Analogous to HCP data, the second task dataset thus incorporated 1404 labeled, gray-matter masked, and z-scored activity maps from 18 diverse tasks acquired in 78 participants.*
    * When it is clear that they are talking about multiple datasets, if it is ambiguous, do mark the dataset.
      · *However, the gain in accuracy is more substantial for the* *SQF datasets* *as compared to the* *COMPAS and Adult datasets.*
      · *In our experiments, we used two standard TREC collections: the first collection (called* *Robust04* *) consists of over 500,000 news articles from different news agencies, which are available in TREC Disks 4 and 5 (excluding Congressional Records).*
    * When an abbreviation is introduced for the article only.
      · *The second collection (called ClueWeb) that we used is* *ClueWeb09 Category B*, *a large-scale web collection with over 50 million English documents, which is considered a heterogeneous collection.*
    * When a dataset is made specifically for the article.
      · *We first consider modeling a segment of the pseudo periodic synthetic dataset.*
    * When no keywords (such as dataset, database, etc.) follow after a (set of) word(s), but the sentence makes clear that the word(s) indeed does refer to a dataset.
      · *We use the* *AOL query log* *[12] in our experiments and preprocess the dataset following [8].*
      · *We evaluate the FS-RNN on two character level language modeling data sets,* *Penn Treebank* *and* *Hutter Prize Wikipedia*, *where we improve state-of-the-art results to 1.19 and 1.25 bits-per-character (BPC), respectively.*

Appendix B.4. What to Annotate †:

- Mark as much information as possible, so include keywords, such as dataset, training set, etc.
  - This includes version numbers.
  - However, do not include words such as 'the'.
- When an abbreviation follows a dataset name or divides the name, mark this as one. See examples below:
  - *Several recent image models were evaluated on small image patches sampled from the Berkeley segmentation dataset (BSDS300) [25].*
  - *Labeled Faces in the Wild (LFW) database [12] is widely used for face recognition and verification benchmark.*
- To determine whether an adjective is part of the dataset name, try to imagine if the meaning of the term would change if the adjective was removed. If this is the case, include the adjective; otherwise, do not include the adjective.
  - *Table 1: NLL scores in the test data for the binarized MNIST dataset .*
  - *(b) Weights W (filters) learned by LeGrad when training an SBN with H = 200 units in the full MNIST training set .*
- If one term contains information that is needed for the other term (ellipses), mark them as one.
  - *Tables 3 and 4 show the average precision of 20 categories and MAP on the PASCAL VOC 2007 and 2012 testing set , respectively.*
  - *State-of-the-art performance on the ASSISTments benchmark and Khan dataset .*
  - *This is equivalent to finding a transport map ϕ from random noises with distribution pX (e.g., Gaussian distribution or uniform distribution) to the underlying population distribution pY of the genuine sample, e.g., the MNIST or the ImageNet dataset .*
  - *For the Blizzard and Accent datasets , we process the data so that each sample duration is 0.5 s (the sampling frequency used is 16 kHz.)*
- If there is a preposition in the term, judge if it is a part of the term or if it splits terms.
  - *For conditional density estimation, we use the MNIST dataset of handwritten digits [17] and the CIFAR-10 dataset of natural images [14].*
  - *The Twitter Part of Speech dataset [4] contains 1827 tweets annotated with 25 POS tags.*
- When the name is divided by references, mark the sentences as if the references are not there.
  - *The Allstate Insurance Claim [27] and the Flight Delay [28] datasets both contain a lot of one-hot coding features.*
- If the marked part contains a spelling error, this has to be fixed (as this cannot be done in the annotating environment, this has to be done after, so make sure to notate the sentences that need to be adjusted). Mark the sentence as if there is no spelling error.

† Based on the ACL RD-TEC Annotation Guideline, version 2.6 [45].

Appendix C. Rule-Based Score

Table A1. Rule-based score on train and zero-shot sets.

Set	Evaluation	Precision	Recall	F1
Test set	B-tags	0.81	0.75	0.77
	I-tags	0.79	0.72	0.76
	Partial-match	0.81	0.71	0.76
	Exact-match	0.76	0.67	0.71
Zero-shot set	B-tags	0.90	0.76	0.82
	I-tags	0.88	0.76	0.82
	Partial-match	0.92	0.78	0.85
	Exact-match	0.88	0.75	0.81

Appendix D. More in Depth Results

Table A2. RQ1, overall performances: cross-validation scores.

Models	Evaluation	Precision		Recall		F1-Score	
		CV-Score	SD	CV-Score	SD	CV-Score	SD
Rule-based	B-tags	0.82	0.01	0.72	0.01	0.77	0.01
	I-tags	0.79	0.01	0.76	0.02	0.77	0.02
	Partial-match	0.81	0.01	0.72	0.02	0.81	0.01
	Exact-match	0.77	0.01	0.68	0.02	0.72	0.01
CRF	B-tags	0.89	0.00	0.65	0.03	0.75	0.02
	I-tags	0.86	0.00	0.72	0.02	0.78	0.02
	Partial-match	0.89	0.00	0.65	0.03	0.75	0.02
	Exact-match	0.85	0.01	0.62	0.03	0.72	0.02
BiLSTM	B-tags	0.86	0.01	0.75	0.01	0.80	0.00
	I-tags	0.72	0.01	0.74	0.01	0.73	0.01
	Partial-match	0.73	0.01	0.72	0.01	0.73	0.00
	Exact-match	0.67	0.01	0.66	0.01	0.67	0.01
BiLSTM-CRF	B-tags	0.82	0.02	0.79	0.03	0.80	0.02
	I-tags	0.78	0.02	0.76	0.03	0.77	0.03
	Partial-match	0.77	0.03	0.76	0.03	0.77	0.02
	Exact-match	0.72	0.03	0.71	0.04	0.72	0.03
BERT	B-tags	0.82	0.01	0.85	0.02	0.84	0.01
	I-tags	0.78	0.01	0.81	0.01	0.80	0.01
	Partial-match	0.79	0.01	0.83	0.01	0.81	0.01
	Exact-match	0.75	0.01	0.79	0.02	0.77	0.01
SciBERT	B-tags	0.83	0.01	0.85	0.02	0.84	0.01
	I-tags	0.79	0.01	0.82	0.01	0.81	0.01
	Partial-match	0.80	0.01	0.84	0.01	0.82	0.01
	Exact-match	0.76	0.02	0.80	0.02	0.78	0.02

Table A3. RQ2, domain adaptability between corpora.

Models	Evaluation	NIPS			SDM			SIGIR			VISION		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
Rule-based	B-tags	0.82	0.72	0.77	0.75	0.73	0.74	0.79	0.77	0.78	0.89	0.68	0.77
	I-tags	0.80	0.75	0.77	0.75	0.75	0.75	0.78	0.77	0.78	0.83	0.75	0.79
	Partial-match	0.81	0.71	0.75	0.77	0.74	0.75	0.80	0.78	0.79	0.86	0.67	0.75
	Exact-match	0.77	0.67	0.72	0.71	0.69	0.70	0.76	0.74	0.75	0.82	0.63	0.71
CRF	B-tags	0.89	0.61	0.72	0.87	0.67	0.76	0.87	0.64	0.73	0.92	0.57	0.70
	I-tags	0.86	0.66	0.74	0.85	0.74	0.79	0.90	0.65	0.76	0.94	0.69	0.79
	Partial-match	0.89	0.61	0.72	0.88	0.67	0.76	0.87	0.64	0.74	0.94	0.58	0.71
	Exact-match	0.84	0.57	0.68	0.84	0.64	0.73	0.83	0.61	0.71	0.90	0.55	0.68
BiLSTM	B-tags	0.86	0.67	0.75	0.86	0.67	0.75	0.85	0.68	0.75	0.85	0.67	0.75
	I-tags	0.69	0.70	0.69	0.70	0.70	0.70	0.68	0.69	0.69	0.69	0.71	0.70
	Partial-match	0.72	0.64	0.68	0.72	0.64	0.68	0.71	0.64	0.67	0.72	0.64	0.68
	Exact-match	0.66	0.59	0.62	0.66	0.59	0.62	0.64	0.59	0.61	0.65	0.59	0.62
BiLSTM-CRF	B-tags	0.87	0.57	0.69	0.86	0.56	0.68	0.87	0.60	0.71	0.86	0.60	0.71
	I-tags	0.67	0.57	0.62	0.66	0.56	0.61	0.67	0.62	0.64	0.66	0.63	0.64
	Partial-match	0.69	0.53	0.60	0.68	0.53	0.59	0.71	0.57	0.64	0.70	0.58	0.63
	Exact-match	0.61	0.47	0.53	0.60	0.46	0.52	0.63	0.51	0.57	0.63	0.52	0.57
BERT	B-tags	0.84	0.81	0.83	0.85	0.86	0.85	0.78	0.86	0.82	0.80	0.79	0.80
	I-tags	0.78	0.78	0.78	0.81	0.84	0.82	0.78	0.83	0.80	0.74	0.74	0.74
	Partial-match	0.80	0.80	0.80	0.81	0.84	0.83	0.74	0.84	0.79	0.77	0.76	0.77
	Exact-match	0.76	0.75	0.75	0.78	0.80	0.79	0.71	0.81	0.76	0.70	0.71	0.71
SciBERT	B-tags	0.84	0.83	0.83	0.85	0.86	0.86	0.77	0.88	0.82	0.81	0.80	0.81
	I-tags	0.78	0.79	0.78	0.81	0.85	0.83	0.77	0.84	0.80	0.73	0.75	0.74
	Partial-match	0.80	0.82	0.81	0.82	0.85	0.83	0.74	0.86	0.80	0.77	0.78	0.78
	Exact-match	0.76	0.77	0.76	0.78	0.82	0.80	0.71	0.83	0.76	0.71	0.72	0.71

Table A4. RQ4, testing on a real ratio (positive vs. negative) test set.

Model	Evaluation	Precision	Recall	F1
CRF	B-tags	0.72	0.66	0.69
	I-tags	0.67	0.73	0.70
	Partial-match	0.72	0.66	0.69
	Exact-match	0.69	0.63	0.66
BiLSTM	B-tags	0.65	0.12	0.20
	I-tags	0.39	0.23	0.29
	Partial-match	0.37	0.19	0.26
	Exact-match	0.29	0.15	0.20
BiLSTM-CRF	B-tags	0.57	0.12	0.20
	I-tags	0.46	0.30	0.36
	Partial-match	0.49	0.29	0.37
	Exact-match	0.38	0.23	0.29
BERT	B-tags	0.59	0.91	0.72
	I-tags	0.69	0.89	0.78
	Partial-match	0.57	0.91	0.70
	Exact-match	0.55	0.88	0.68
SciBERT	B-tags	0.66	0.91	0.77
	I-tags	0.73	0.89	0.80
	Partial-match	0.65	0.91	0.76
	Exact-match	0.63	0.88	0.73

Table A5. RQ5, differences between training on the golden standard or the weakly annotated data.

Training Set	Evaluation	Test Set			Zero-Shot		
		P	R	F1	P	R	F1
Train set	B-tags	0.87	0.87	0.87	0.87	0.79	0.82
	I-tags	0.82	0.84	0.83	0.84	0.78	0.81
	Partial-match	0.82	0.84	0.83	0.84	0.78	0.81
	Exact-match	0.79	0.82	0.80	0.79	0.74	0.76
SSC	B-tags	0.77	0.71	0.74	0.83	0.70	0.76
	I-tags	0.74	0.73	0.73	0.80	0.69	0.74
	Partial-match	0.76	0.71	0.73	0.79	0.69	0.73
	Exact-match	0.70	0.65	0.67	0.72	0.63	0.67

Table A6. RQ5, training on three different distributions of real and 'weak' examples.

Training Set	Evaluation	Test Set			Zero-Shot		
		P	R	F1	P	R	F1
Train + SSC	B-tags	0.84	0.83	0.84	0.84	0.77	0.80
	I-tags	0.79	0.80	0.80	0.86	0.79	0.82
	Partial-match	0.81	0.82	0.81	0.84	0.78	0.81
	Exact-match	0.76	0.77	0.76	0.80	0.74	0.77
SSC + Train	B-tags	0.84	0.81	0.83	0.84	0.79	0.81
	I-tags	0.78	0.79	0.78	0.85	0.79	0.82
	Partial-match	0.81	0.81	0.81	0.82	0.78	0.80
	Exact-match	0.76	0.75	0.76	0.77	0.74	0.75
SSC & Train shuffled	B-tags	0.83	0.83	0.83	0.85	0.81	0.83
	I-tags	0.77	0.81	0.79	0.84	0.81	0.83
	Partial-match	0.79	0.82	0.80	0.84	0.81	0.83
	Exact-match	0.74	0.77	0.75	0.79	0.76	0.78

Table A7. RQ5, adding only positive 'weak' examples.

Training Set	Evaluation	Test Set			Zero-Shot		
		P	R	F1	P	R	F1
Train set & positive examples	B-tags	0.83	0.86	0.85	0.84	0.79	0.82
	I-tags	0.78	0.83	0.80	0.83	0.80	0.82
	Partial-match	0.80	0.85	0.83	0.84	0.81	0.82
	Exact-match	0.76	0.81	0.78	0.78	0.76	0.77

Appendix E. From PDF to Machine Readable Format

We initially extracted the text from the PDFs using pdftotext, an open-source command-line utility for converting PDF files to plain text files [60]. We later discovered a much more powerful tool for preprocessing scientific papers for scientific information extraction, named GROBID. GROBID is a machine learning library for extracting, parsing and re-structuring raw documents, such as PDFs, into structured XML/TEI encoded documents with a particular focus on technical and scientific publications [39]. The advantage of using GROBID over pdf2text is that GROBID is able to extract more information than pdf2text. The core features as described by GROBID include the following:

1. Header extraction and parsing (e.g., title, abstract, authors, affiliations, keywords, etc.);
2. References extraction and parsing (including DOI identifiers);
3. Citation contexts recognition and linking (including URL extraction);
4. Full text extraction and structuring (e.g., paragraph, section titles, reference callout, figure, table, and foot notes).

For each conference, we parsed the publications available to us and extracted the following information from the GROBID conversion output: *article ID, conference, title of the publication, raw text, sections (number, title and index of starting character), list of author names, references* and *status of publication*.

References

1. Brickley, D.; Burgess, M.; Noy, N. Google Dataset Search: Building a Search Engine for Datasets in an Open Web Ecosystem. In Proceedings of the World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 1365–1375. [CrossRef]
2. Kratz, J.E.; Strasser, C. Researcher perspectives on publication and peer review of data. *PLoS ONE* **2015**, *10*, e0117619. [CrossRef]
3. Ghavimi, B.; Mayr, P.; Vahdati, S.; Lange, C. Identifying and Improving Dataset References in Social Sciences Full Texts. *arXiv* **2016**, arXiv:1603.01774.
4. Zeng, T.; Wu, L.; Bratt, S.; Acuna, D.E. Assigning credit to scientific datasets using article citation networks. *arXiv* **2020**, arXiv:2001.05917.
5. Mathiak, B.; Boland, K. Challenges in matching dataset citation strings to datasets in social science. *D-Lib Mag.* **2015**, *21*, 23–28. [CrossRef]
6. Prasad, A.; Si, C.; Kan, M.Y. Dataset Mention Extraction and Classification. In Proceedings of the Workshop on Extracting Structured Knowledge from Scientific Publications, Minneapolis, MN, USA, 19–26 June 2019; Association for Computational Linguistics: Minneapolis, MN, USA, 2019; pp. 31–36. [CrossRef]
7. Luan, Y.; He, L.; Ostendorf, M.; Hajishirzi, H. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. *arXiv* **2018**, arXiv:1808.09602.
8. Ghavimi, B.; Mayr, P.; Lange, C.; Vahdati, S.; Auer, S. A semi-automatic approach for detecting dataset references in social science texts. *Inf. Serv. Use* **2016**, *36*, 171–187. [CrossRef]
9. Yao, R.; Hou, L.; Ye, Y.; Wu, O.; Zhang, J.; Wu, J. Method and Dataset Mining in Scientific Papers. *arXiv* **2019**, arXiv:1911.13096.
10. Zhao, H.; Luo, Z.; Feng, C.; Zheng, A.; Liu, X. A Context-based Framework for Modeling the Role and Function of On-line Resource Citations in Scientific Literature. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, Hong Kong, China, 3–7 November 2019; Association for Computational Linguistics: Hong Kong, China, 2019; pp. 5206–5215. [CrossRef]
11. Kim, H.; Park, K.; Park, S.H. Rich Context Competition: Extracting Research Context and Dataset Usage Information from Scientific Publications. Available online: https://rokrokss.com/assets/cv/rcc09.pdf (accessed on 1 June 2020).
12. Erera, S.; Shmueli-Scheuer, M.; Feigenblat, G.; Peled Nakash, O.; Boni, O.; Roitman, H.; Cohen, D.; Weiner, B.; Mass, Y.; Rivlin, O.; et al. A Summarization System for Scientific Documents. *arXiv* **2019**, arXiv:1908.11152.
13. Hou, Y.; Jochim, C.; Gleize, M.; Bonin, F.; Ganguly, D. Identification of Tasks, Datasets, Evaluation Metrics, and Numeric Scores for Scientific Leaderboards Construction. *arXiv* **2019**, arXiv:1906.09317.
14. Duck, G.; Nenadic, G.; Brass, A.; Robertson, D.L.; Stevens, R. bioNerDS: Exploring bioinformatics' database and software use through literature mining. *BMC Bioinform.* **2013**, *14*, 194. [CrossRef]
15. Beltagy, I.; Lo, K.; Cohan, A. SciBERT: A Pretrained Language Model for Scientific Text. *arXiv* **2019**, arXiv:1903.10676.
16. Gábor, K.; Buscaldi, D.; Schumann, A.K.; QasemiZadeh, B.; Zargayouna, H.; Charnois, T. SemEval-2018 Task 7: Semantic Relation Extraction and Classification in Scientific Papers. In Proceedings of the 12th International Workshop on Semantic Evaluation, New Orleans, Louisiana, 5–6 June 2018; pp. 679–688. [CrossRef]
17. Casillas, A.; Ezeiza, N.; Goenaga, I.; Pérez, A.; Soto, X. Measuring the effect of different types of unsupervised word representations on Medical Named Entity Recognition. *Int. J. Med. Inform.* **2019**, *129*, 100–106. [CrossRef]
18. Guo, H.; Zhu, H.; Guo, Z.; Zhang, X.; Wu, X.; Su, Z. Domain Adaptation with Latent Semantic Association for Named Entity Recognition. In Proceedings of the Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Boulder, CO, USA, 31 May 2009–5 June 2009; Association for Computational Linguistics: Boulder, CO, USA, 2009; pp. 281–289.
19. Lee, J.; Kim, H.; Lee, J.; Yoon, S. Transfer learning for deep learning on graph-structured data. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
20. Zhang, L. Transfer Adaptation Learning: A Decade Survey. *arXiv* **2019**, arXiv:1903.04687.
21. Song, Y.; Yi, E.; Kim, E.; Lee, G.G.; Park, S.J. POSBIOTM-NER: A machine learning approach for bio-named entity recognition. In Proceedings of the Workshop on Critical Assessment of Text Mining Methods in Molecular Biology, Granada, Spain, 28–31 March 2004.
22. Augenstein, I.; Derczynski, L.; Bontcheva, K. Generalisation in Named Entity Recognition: A quantitative analysis. *Comput. Speech Lang.* **2017**, *44*, 61–83. [CrossRef]
23. Kim, J.; Kim, J. The impact of imbalanced training data on machine learning for author name disambiguation. *Scientometrics* **2018**, *117*, 511–526. [CrossRef]
24. Kurczab, R.; Smusz, S.; Bojarski, A.J. The influence of negative training set size on machine learning-based virtual screening. *J. Cheminform.* **2014**, *6*, 32. [CrossRef] [PubMed]
25. Li, X.L.; Liu, B.; Ng, S.K. Negative Training Data Can Be Harmful to Text Classification. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, Cambridge, MA, USA, 9–11 October 2010; Association for Computational Linguistics: Cambridge, MA, USA, 2010; pp. 218–228.

26. Huang, X.; Dong, L.; Boschee, E.; Peng, N. Learning A Unified Named Entity Tagger From Multiple Partially Annotated Corpora For Efficient Adaptation. *arXiv* **2019**, arXiv:1909.11535.
27. Khongtum, O.; Promrit, N.; Waijanya, S. The Entity Recognition of Thai Poem Compose by Sunthorn Phu by Using the Bidirectional Long Short Term Memory Technique. In Proceedings of the International Conference on Multi-disciplinary Trends in Artificial Intelligence, Kuala Lumpur, Malaysia, 17–19 November 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 97–108.
28. Li, Z.; Zhang, Q.; Liu, Y.; Feng, D.; Huang, Z. Recurrent neural networks with specialized word embedding for chinese clinical named entity recognition. In Proceedings of the Evaluation Task at the China Conference on Knowledge Graph and Semantic Computing, Chengdu, China, 26–29 August 2017; pp. 55–60.
29. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805.
30. Tchoua, R.; Ajith, A.; Hong, Z.; Ward, L.; Chard, K.; Audus, D.; Patel, S.; de Pablo, J.; Foster, I. Towards hybrid human-machine scientific information extraction. In Proceedings of the 2018 New York Scientific Data Summit, New York, NY, USA, 6–8 August 2018; pp. 1–3. [CrossRef]
31. Humphreys, K.; Demetriou, G.; Gaizauskas, R. Bioinformatics applications of information extraction from scientific journal articles. *J. Inf. Sci.* **2000**, *26*, 75–85. [CrossRef]
32. Liddy, E.D. The discourse-level structure of empirical abstracts: An exploratory study. *Inf. Process. Manag.* **1991**, *27*, 55–81. [CrossRef]
33. Mohit, B. Named entity recognition. In *Natural Language Processing of Semitic Languages*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 221–245.
34. Chiticariu, L.; Krishnamurthy, R.; Li, Y.; Reiss, F.; Vaithyanathan, S. Domain Adaptation of Rule-Based Annotators for Named-Entity Recognition Tasks. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, Cambridge, MA, USA, 9–11 October 2010; Association for Computational Linguistics: Cambridge, MA, USA, 2010; pp. 1002–1012.
35. Sterckx, L. Methods for Efficient Supervision in Natural Language Processing. Ph.D. Thesis, Ghent University, Ghent, Belgium, 2018.
36. Fries, J.A.; Varma, P.; Chen, V.S.; Xiao, K.; Tejeda, H.; Saha, P.; Dunnmon, J.; Chubb, H.; Maskatia, S.; Fiterau, M.; et al. Weakly supervised classification of aortic valve malformations using unlabeled cardiac MRI sequences. *Nat. Commun.* **2019**, *10*, 1–10. [CrossRef] [PubMed]
37. Soni, A.; Viswanathan, D.; Pachaiyappan, N.; Natarajan, S. A Comparison of Weak Supervision methods for Knowledge Base Construction. In Proceedings of the 5th Workshop on Automated Knowledge Base Construction, San Diego, CA, USA, 17 June 2016; Association for Computational Linguistics: San Diego, CA, USA, 2016; pp. 97–102. [CrossRef]
38. Klinger, R.; Tomanek, K. Classical Probabilistic Models and Conditional Random Fields. Available online: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.645.5543&rep=rep1&type=pdf (accessed on 1 June 2020).
39. Lopez, P. GROBID. 2008–2020. Available online: https://github.com/kermitt2/grobid (accessed on 1 June 2020).
40. Colón-Ruiz, C.; Segura-Bedmar, I. Protected Health Information Recognition byBiLSTM-CRF. Available online: http://ceur-ws.org/Vol-2421/MEDDOCAN_paper_6.pdf (accessed on 1 June 2020).
41. Wunnava, S.; Qin, X.; Kakar, T.; Rundensteiner, E.A.; Kong, X. Bidirectional LSTM-CRF for adverse drug event tagging in electronic health records. In Proceedings of the 1st International Workshop on Medication and Adverse Drug Event Detection, 4 May 2018; pp. 48–56.
42. Simoes, G.; Galhardas, H.; Coheur, L. Information Extraction tasks: A survey. In Proceedings of the INForum 2009-Simpósio de Informática, Lisboa, Portugal, 10–11 September 2009.
43. Śniegula, A.; Poniszewska-Marańda, A.; Chomatek, Ł. Towards the Named Entity Recognition Methods in Biomedical Field. In Proceedings of the 46th International Conference on Current Trends in Theory and Practice of Informatics, Limassol, Cyprus, 20–24 January 2020; pp. 375–387.
44. Correia, G.M.; Niculae, V.; Martins, A.F.T. Adaptively Sparse Transformers. *arXiv* **2019**, arXiv:cs.CL/1909.00015, e-prints.
45. Schumann, A.K.; Qasemi Zadeh, B. The ACL RD-TEC Annotation Guideline: A Reference Dataset for the Evaluation of Automatic Term Recognition and Classification. *Tech. Rep.* **2015**. [CrossRef]
46. Nichols, T.R.; Wisner, P.M.; Cripe, G.; Gulabchand, L. Putting the kappa statistic to use. *Qual. Assur. J.* **2010**, *13*, 57–61. [CrossRef]
47. spaCy. Tokenizer. Available online: https://spacy.io/api/tokenizer (accessed on 1 June 2020).
48. NLTK. Natural Language Toolkit. Available online: https://www.nltk.org/index.html (accessed on 1 June 2020).
49. Github. A Python Framework for Sequence Labeling Evaluation (Named-Entity Recognition, Pos Tagging, etc.). 2019. Available online: https://github.com/chakki-works/seqeval (accessed on 1 June 2020).
50. Github. Full Named-Entity (i.e., Not Tag/Token) Evaluation Metrics Based on SemEval'13. 2019. Available online: https://github.com/ivyleavedtoadflax/nervaluate (accessed on 1 June 2020).
51. spaCy. Rule-Based Matching. Available online: https://spacy.io/usage/rule-based-matching/ (accessed on 1 June 2020).
52. Sklearn. Sklearn−Crfsuite. Available online: https://sklearn-crfsuite.readthedocs.io/en/latest/ (accessed on 1 June 2020).
53. Depends on the Definition Guide to Sequence Tagging with Neural Networks. 2017. Available online: https://www.depends-on-the-definition.com/guide-sequence-tagging-neural-networks-python/ (accessed on 1 June 2020).
54. Github. Scikit-Learn Wrapper to Finetune BERT 2019. Available online: https://github.com/charles9n/bert-sklearn (accessed on 1 June 2020).

55. Chowdhury, M.F.M.; Lavelli, A. Assessing the practical usability of an automatically annotated corpus. In Proceedings of the 5th Linguistic Annotation Workshop, Portland, OR, USA, 23–24 June 2011.
56. Dutta, S.; Weikum, G. Cross-document co-reference resolution using sample-based clustering with knowledge enrichment. *Trans. Assoc. Comput. Linguist.* **2015**, *3*, 15–28. [CrossRef]
57. Stevenson, A. *Oxford Dictionary of English*; Oxford University Press: Oxford, UK, 2010.
58. Dictionary, O.L. Data-Set. 2019. Available online: https://www.oxfordlearnersdictionaries.com/definition/english/data-set (accessed on 1 June 2020).
59. NIPS. NIPS Proceedings. Available online: https://papers.nips.cc/ (accessed on 1 June 2020).
60. Palmer, J. pdftotext. 2020. Available online: https://github.com/jalan/pdftotext (accessed on 1 June 2020).