Cold is a Disease and D-cold is a Drug: Identifying Biological Types of Entities in the Biomedical Domain

Suyash Sangwan[1], Raksha Sharma[2], Girish Palshikar[3] and Asif Ekbal[1]

[1]Indian Institute of Technology Patna, India
[2]Indian Institute of Technology, Roorkee, India
[3]TCS Innovation Labs, India
[1]{suyash.mtmc17,asif}@iitp.ac.in
[2]rakshasharma.fcs@iitr.ac.in
[3]gk.palshikar@tcs.com

Abstract. Automatically extracting different types of knowledge from authoritative biomedical texts, *e.g.*, scientific medical literature, electronic health records *etc.*, and representing it in a computer analyzable as well as human-readable form is an important but challenging task. One such knowledge is identifying entities with their biological types in the biomedical domain.
In this paper, we propose a system which extracts end-to-end entity mentions with their biological types from a sentence. We consider 7 interrelated tags for biological types *viz.,* gene, biological-process, molecular-function, cellular-component, protein, disease, drug. Our system employs an automatically created biological ontology and implements an efficient matching algorithm for end-to-end entity extraction. We compare our approach with a Noun-based entity extraction system (*baseline*) as well as we show a significant improvement over standard entity extraction tools, *viz.,* Stanford-NER, Stanford-OpenIE.

1 Introduction

An enormous amount of biomedical data have been generated and collected at an unprecedented speed and scale. For example, the application of electronic health records (EHRs) is documenting large amounts of patient data. However, retrieving and processing this information is very difficult due to the lack of formal structure in the natural language used in these documents. Therefore we need to build systems which can automatically extract the information from the biomedical text which holds the promise of easily consolidating large amounts of biological knowledge in computer or human accessible form. Ability to query and use such extracted knowledge-bases can help scientists, doctors and other users in performing tasks such as question-answering, diagnosis and identifying opportunities for the new research.

Automatic identification of entities with their biological types is a complex task due to the domain-specific occurrences of entities. Consider the following example to understand the problem well.

- Input Sentence: *Twenty courses of 5-azacytidine (5-Aza) were administrated as maintenance therapy after induction therapy with daunorubicin and cytarabine.*

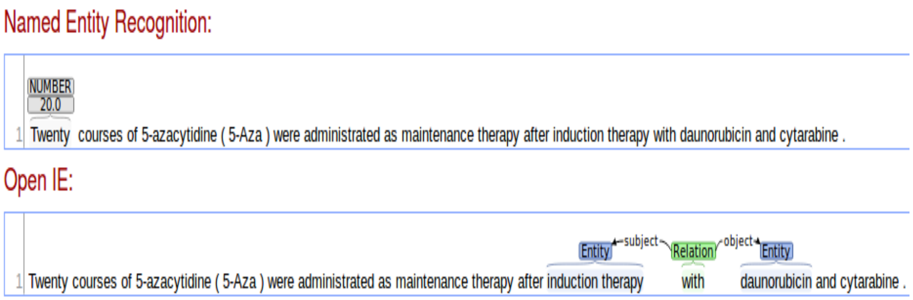
- Entities Found = {*5-azacytidine, daunorubicin, cytarabine*}

- Biological types for the Entities = {*drug, drug, drug*}

All the three extracted entities in the example are specific to the biomedical domain having biological type *drug*. Hence, an entity extraction tool trained on generic data will not be able to capture these entities. Figure 1 shows the entities tagged by Stanford-NER and Stanford-OpenIE tools. Stanford-NER fails to tag any of the entity, while Stanford-OpenIE is able to tag *daunorubicin*.

In this paper, we propose an approach which uses the biomedical ontology to extract end-to-end entity mentions with their biological types from a sentence in the biomedical domain. *By end-to-end we mean that correctly identify the boundary of each entity mention*. We consider 7 interrelated tags for biological types *viz.,* gene, biological-process, molecular-function, cellular-component, protein, disease, drug. They together form a complete biological system, where one biological type is the cause or effect of another biological type. The major contribution of this research is as follows.

1. *Ontology in the biomedical domain:* Automatic creation of ontology having biological entities with their biological types.
2. *Identifying end-to-end entities with their biological types:* We have implemented an efficient matching algorithm, named it All Subsequences Entity Match (ASEM). It is able to extract entities with their biological types from a sentence using ontology. ASEM is also able to tag entities which are the subsequence of another entity. For example, for *mammalian target of rapamycin*, our system detects two entities {*mammalian target of rapamycin, rapamycin*} with biological types {*protein, drug*} respectively.



Named Entity Recognition:

1 Twenty courses of 5-azacytidine (5-Aza) were administrated as maintenance therapy after induction therapy with daunorubicin and cytarabine .

Open IE:

1 Twenty courses of 5-azacytidine (5-Aza) were administrated as maintenance therapy after induction therapy with daunorubicin and cytarabine .

Fig. 1: Entity tagging by Stanford-NER and Stanford-OpenIE

Since nouns are the visible candidates for being entities, we consider a Noun-based entity extraction system as a baseline. This system uses NLTK POS tagger for tagging the words with POS tags. In addition, we compare performance of our ASEM-based approach with Stanford-NER[1] and Stanford-OpenIE[2]. The rest of the paper is organized as follows. Section 2 describes the related work. Section 3 gives a description of the dataset used. Section 4 presents the ontology creation details and the ASEM algorithm. Section 5 provides experimental setup and results and Section 6 concludes the paper.

2 Related Work

Entity Extraction has been a widely studied area of research in NLP. There have been attempts for both supervised as well as unsupervised techniques for entity extraction task [1]. Etzioni et al., (2005) [2] proposed an unsupervised approach to extract named entities from the Web. They built a system KNOWITALL, which is a domain-independent system that extracts information from the Web in an unsupervised and open-ended manner. KNOWITALL introduces a novel, generate-and-test architecture that extracts information in two stages. KNOWITALL utilizes a set of eight domain-independent extraction patterns to generate candidate facts. Baluja et al., (2000) [3] presented a machine learning approach for building an efficient and accurate name spotting system. They described a system that automatically combines weak evidence from different, easily available sources: partsofspeech tags, dictionaries, and surfacelevel syntactic information such as capitalization and punctuation. They showed that the combination of evidence through standard machine learning techniques yields a system that achieves performance equivalent to the best existing hand-crafted approach. Carreras et al., (2002) [4] presented a Named Entity Extraction (NEE) problem as two tasks, recognition (NER) and classification (NEC), both the tasks were performed sequentially and independently with separate modules. Both modules are machine learning based systems, which make use of binary AdaBoost classifiers. Cross-lingual techniques are also developed to build an entity recognition system in a language with the help of another resource-rich language [5–11].

There are a few instances of use of already existing ontology or creation of a new ontology for entity extraction task. Cohen and Sarawagi, (2004) [12] considered the problem of improving named entity recognition (NER) systems by using external dictionaries. More specifically, they extended state-of-the-art NER systems by incorporating information about the similarity of extracted entities to entities in an external dictionary. Textpresso's which is a tool by Muller et al., (2004) [13] has two major elements, it has a collection of the full text of scientific articles split into individual sentences and the implementation of categories of terms for which a database of articles and individual sentences can be searched. The categories are classes of biological concepts (*e.g.*, gene, allele, cell or cell group, phenotype, *etc.*) and classes that relate two objects (*e.g.*, association, regulation, *etc.*) or describe one (*e.g.*, biological process, *etc.*). Together they form a catalog of types of objects and concepts called an ontology. Wang et al., (2009) [14] used approximate dictionary matching with edit distance constraints. Their solution was based on an improved neighborhood generation method employing partitioning and prefix pruning techniques. They showed that their entity recognition system

[1] Available at: `https://nlp.stanford.edu/software/CRF-NER.shtml`
[2] Available at: `https://nlp.stanford.edu/software/openie.html`

was able to capture typographical or orthographical errors, both of which are common in entity extraction tasks yet may be missed by token-based similarity constraints.

There are a few instances of entity extraction in the biomedical domain [15–17]. Takeuchi and Collier (2005) [18] applied Support Vector Machine for the identification and semantic annotation of scientific and technical terminology in the domain of molecular biology. This illustrates the extensibility of the traditionally named entity task to special domains with large-scale terminologies such as those in medicine and related disciplines. More recently, Joseph et al., (2012) [19] built a search engine dedicated to the biomedical domain, they also populated a dictionary of domain-specific entities. In this paper, we have proposed an unsupervised approach, which first generates a domain-specific ontology and then performs all subsequence matches against the ontology entries for entity extraction in the biomedical domain.

3 Dataset

To evaluate the performance of our algorithm and other approaches, we asked an expert to manually annotate a dataset of 50 abstracts (350 sentences) of Leukemia-related papers from PubMed [20] having cause-effect relations. We obtained 231 biological entities with their biological types. Below is an example from the manually tagged output. Entities are enclosed in curly brackets and their types are attached using '_' symbol.

{6-Mercaptopurine}_drug (6-MP_drug) is one of the main components for the treatment of childhood {acute lymphoblastic leukemia}_disease (ALL_disease).

To observe the performance of our system on a large corpus, we used an untagged corpus of $10,000$ documents given by Sharma et al., (2018) [20]. They downloaded $10,000$ abstracts of Leukemia-related papers from PubMed using the Biopython library with Entrez package. They used this dataset ($89,947$ sentences having $1,935,467$ tokens) to identify causative verbs in the biomedical domain.

4 Approach

In this paper, we present an approach to identify end-to-end biological entities with their biological types in a sentence using automatically created ontology. The following sections 4.1 and 4.2 elaborate the process of ontology creation and ASEM algorithm.

4.1 Ontology Creation

We automatically built an ontology for 7 biological types, *viz.,* gene, biological-process, molecular-function, cellular-component, protein, disease, drug. We referred to various authentic websites having biological entity names with their types.[3] Since direct downloadable links are not available to obtain complete dataset, we built a customized HTML

[3] 1. http://www.geneontology.org/, 2. https://bioportal.bioontology.org/ontologies/DOID, 3. http://browser.planteome.org/amigo/search/ontology?q=\%20regimen

parser to extract biological entities with their types. The selection of the websites for this task is done manually. We obtained an ontology of size $90,567$ with our customized HTML parser.

Joseph et al., (2012) [19] also created a dictionary for biological entities having information about their biological types. They used this dictionary to equip with TPX. TPX is a Web-based PubMed search enhancement tool that enables faster article searching using analysis and exploration features. Their process of creating dictionary from the various sources, has been granted a Japanese patent (JP2013178757). In order to enrich our ontology further, we included entities available in TPX.

Entities	**Biological Type**
Chitobiase	Gene
Reproduction	Biological-process
Acyl binding	Molecular-function
Obsolete repairosome	Cellular-Component
Delphilin	Protein
Acanthocytoses	Disease
Calcimycin	Drug

Table 1: Ontology: Entities-name and Biological-type

We found approx $1,50,000$ new biological entities with their types from Joseph et al., [19] work. The entire ontology is stored in a Hash table format, where *entity name* is the unique key and *biological type* is the value. Table 1 shows a few entries from the ontology used in the paper. Table 2 depicts the total number of entities extracted with respect to each biological type.

Biological Type	**No. of Entities**
Gene	1,79,591
Biological-process	30,695
Molecular-function	11,936
Cellular-component	4,376
Protein	1,16,125
Disease	74,470
Drug	52,923

Table 2: Ontology Statistics

4.2 Algorithm: Identify Entity with its Biological Type

Nouns are explicitly visible candidates for being a biological entity, we designed a Noun-based entity extraction system. Performance of this system completely depends on the POS tagger, which assigns NOUN tag. The Noun-based system is not able to identify end-to-end entities or the correct entity boundary. Our ASEM-based system is able to find the boundary of an entity in a sentence without POS tag information.

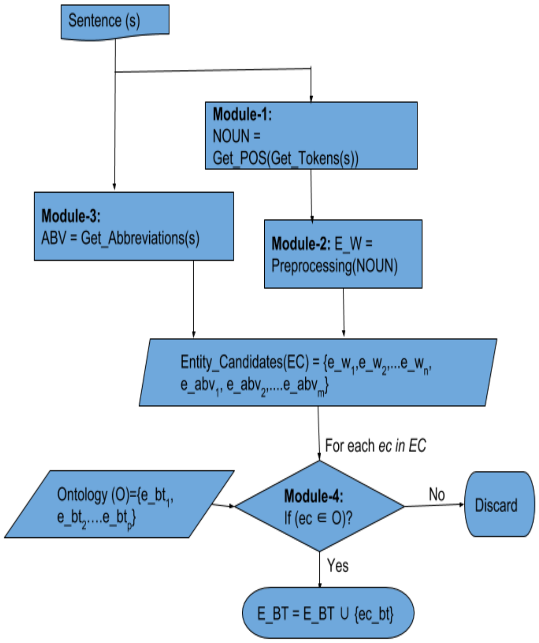


Fig. 2: Work flow of the Noun-based System

Noun-based System The system comprises 4 modules. Figure 2 depicts the workflow of the Noun-based system. The description of the modules is as follows.

Module-1: POS Tagging

We used NLTK POS tagger to tokenize and assign POS tags to words. Words which are tagged as Noun are considered as candidates for being a biological entity. The tagger is trained on general corpus.[4] We observed that NLTK failed to assign correct tags

[4] We also experimented with Stanford POS tagger, but the performance of this tagger was worse than NLTK tagger for the biological entities.

to many words specific to the biomedical domain. For example, *6-Mercaptopurine* is tagged as Adjective by NLTK, however, it is the name of a medicine used for Leukemia treatment, hence it should be tagged as a noun.

Module-2: Preprocessing
Since NLTK tokenizes and tags many words erroneously, we apply preprocessing on the output produced by Module-1. In the preprocessing step, we removed all single letter words which are tagged as Noun, and words starting and ending with a symbol. In order to reduce the percentage of wrongly tagged words, we removed stop words also. For this purpose, we used a standard list of stop words (very high-frequency words) in the biomedical domain.[5] Below are a few examples of stop words from the list.

{ *Blood, analysis, acid, binding, brain, complex*}

Module-3: Get Abbreviations (Abv)
We observed that there were entries in the ontology for the abbreviation of the entity, but not for the actual entity. To capture such instances, we defined rules to form abbreviations from the words of a sentence. For example, *acute lymphoblastic leukemia* was also represented as *ALL*. In such scenario, if *acute lymphoblastic leukemia* is missing in the ontology, but *ALL* is present, we assign the biological type of *ALL* to *acute lymphoblastic leukemia*

Module-4: Extract Biological Type
This module searches for the Entity Candidate (EC) in the ontology (O). If there is an exact match for the candidate word, the Biological Type (BT) of the entity is extracted. The final outcome of this module is the biological type attached to the entity name.

ASEM-based System All Subsequences Entity Match algorithm finds all subsequences up-to sentence length (n) from the sentence whose entities have to be recognized with biological types.[6] In order to get all possible subsequences, we used *n-gram* package of NLTK.[7] This system doesn't require preprocessing step as it doesn't consider POS tagged words, hence there is no error due to the tagger. Module-3 (Get Abbreviations) of the Noun-based approach is also part of ASEM algorithm as it helps to get the biological type of the entity whose abbreviation is an entry in the ontology, but not the entity itself. If we find an entry in the ontology for any subsequence, we consider the subsequence as a valid biological entity and retrieve the biological type of the entity from the ontology. Algorithm 1 gives the pseudo code of the proposed approach. Table 3 defines the functions and symbols used in Algorithm 1.

[5] Available at: `https://www2.informatik.hu-berlin.de/~hakenber/corpora/medline/wordFrequencies.txt`

[6] Though we obtained subsequences up-to length (n), we observed that there were no entity more than 4 words long.

[7] Available at: `http://www.nltk.org/_modules/nltk/model/ngram.html`

Algorithm 1: Identifying Biological Types of Biological Entities

```
Input: WP = $\{w_B^1, w_B^2, ...w_B^k\}$,
       TPX$_{Dictionary}$,
       $S = \{s_B^1, s_B^2, ....s_B^m\}$,

Output: Entity names with their Entity Types for s∈S
$Ontology$ := $\emptyset$;
for each Web-page $wp \in WP$ do
    $Ontology[Entity_{Name}, Entity_{Type}] := HTML_{Parser}(wp)$
$Ontology[Entity_{Name}, Entity_{Type}] :=$
  $Ontology[Entity_{Name}, Entity_{Type}] \cup TPX_{Dictionary}$
for each sentence $s \in S$ do
    E_BT = $\emptyset$
    $NG$ := n-grams(s), //Getting all subsequence of S
    where $n \in$ {1,2,..,length(s)}
    for each $ng \in NG$ do
        $abv_{ng} := Get\_Abbreviation(ng)$
        if $ng$ in Ontology then  E_BT := E_BT $\cup$ ($ng$,Ontology[$ng$])
        if $abv_{ng}$ in Ontology then  E_BT := E_BT $\cup$ ($ng$,Ontology[$abv_{ng}$])
    Entities in $s$ with their biological types : E_BT
```

5 Experimental Setup and Results

In this paper, we hypothesize that matching of all subsequences of a sentence against automatically created ontology in the biomedical domain can efficiently extract end-to-end entities with their biological types. We compare our ASEM-based system with a Noun-based system (4.2), NER-based system and OpenIE-based system. Named entities are good candidates for being biological entities. We have used Stanford Named Entity Recognizer (NER) to obtain entities. On the other hand, Open information extraction (OpenIE) refers to the extraction of binary relations, from plain text, such as (*Mark Zuckerberg; founded; Facebook*). It assigns *subject* and *object* tags to related arguments. We considered these two arguments as candidates for being an entity. In this paper, we have used Stanford-OpenIE tool. NER and OpenIE are able to extract end-to-end entities, in other words, they are able to tag entities having multiple words. However, they both fail to tag many of the entities which are specific to the biomedical domain (See example in Figure 1). The Algorithm 1 remains the same with NER or OpenIE, except the all subsequences set NG is replaced with the set of entities extracted by NER or OpenIE.

Table 4 shows the results obtained with the 4 different systems, *viz.,* Noun-based, NER-based, OpenIE-based, and ASEM-based (our approach) on test data of 350 sentences having manually annotated 231 entities with their biological types. A True Positive (TP) scenario is when both entity and its type exactly match with the manually tagged entry, else False Positive (FP). A False Negative (FN) scenario is when a manual tagging is there for an entity, but the same is not produced by the system. We have

Symbol	Description
WP	Set of relevant Web-pages
TPX$_{Dictionary}$	Dictionary by Joseph et al., [19]
S	Set of sentences in the Biomedical (B) Domain
$Ontology$	Hash-table having entity-name as key and its biological-type as value
$HTML_{Parser}()$	Extracts entity-name and its value from a HTML page
E_BT	Set of entities tagged with Biological Types (BT)
n-grams()	A function to obtain all subsequences (ng) of $s \in S$
$Get\_Abbreviations()$	A function to generate abbreviation of ng

Table 3: Symbols used in Algorithm 1

System	P	R	F
Noun-based	74.54	35.49	48.09
NER	94.44	7.35	13.65
OpenIE	93.75	12.98	22.81
ASEM	**95.86**	**80.08**	**87.26**

Table 4: Precision (P), Recall (R) and F-score (F) using different approaches in %.

used the same ontology to obtain biological type with all 4 systems of Table 4. Results validate our hypothesis that ASEM-based system is able to obtain a satisfactory level of Precision (P), recall (R) and F-score (F) for this domain-specific task. Though Precision is good for all cases, first three systems fail to score good Recall (R) as they use external NLP tools to extract entities from text.

B-Type	P	R	F
Gene	92	95	92
Biological-process	100	75	86
Molecular-function	86	50	63
Cellular-component	100	10	18
Protein	93	90	91
Disease	100	100	100
Drug	100	100	100

Table 5: Precision, Recall and F-score in % with respect to biological type with ASEM-based system.

Table 5 shows the results obtained with ASEM-based system for each biological type. We obtained a positive Pearson correlation of 0.67 between Recall ('R' column of Table 5) obtained for the biological types and the size ('Entity' column of Table 2) of the ontology. The positive correlation asserts that enriching the ontology further would enhance the performance of our approach.

Error Analysis: In the Noun-based system, where we have considered nouns as candidates for entities, precision is minimum as compared to other approaches. NLTK (or Stanford) POS tagger is not able to correctly tag domain-specific entities like *PI3K/AKT , JNK/STAT etc.* (words having any special character in between), they treat PI3K and AKT as two separate words and assign tags accordingly. Below is an example from the biomedical domain which shows the use of these entities.

> *"Targeted therapies in pediatric leukemia are targeting BCR/ABL, TARA and FLT3 proteins, which activation results in the downstream activation of multiple signaling pathways, including the PI3K/AKT, JNK/STAT, Ras/ERK pathways"*

These random breaks in entities introduced by the POS taggers cause a drop in the overall precision of the system. In addition, the Noun-based system is not able to detect the boundary of the entity. However, the biomedical domain is full of entities constituting multiple words. Hence, the Noun-based system produces a poor F-Score of 48.09%.

The NER-based system which uses Stanford-NER tagger is not breaking words like *BCR/ABL, PI3K/AKT, JNK/STAT, Ras/ERK etc.,* as separate entities, unlike the Noun-based system. Therefore Precision is quite high than the Noun-based system. But due to the generic behavior of Stanford-NER, it is able to extract very few entities. So false negatives increase abruptly and hence recall score drops down to 7.35%. On the other hand, OpenIE-based system considers all subject and object as candidates for entities, therefore there are relatively higher chances to extract the exact entity.

Our ASEM-based approach considers all subsequences of the input sentence as candidates for entities and matches these subsequences against the entries in the ontology. Therefore we are able to get all one-word entities, abbreviations and entities constituting more than one words. Consequently, we obtain a high Recall with our approach. However, the ontology is automatically created from the Web. There are a few entities in our Gold standard dataset, which are not found in the ontology. On the other hand, ontology also contains a few words like *led, has, next*, which are not the biological entities as per our annotator. These lacunae in the ontology cause drop in the P and F score of our system. A positive correlation of 0.67 between Recall and ontology size justifies that enriching the ontology further would enhance the performance of our approach.

6 Conclusion

The biomedical domain is full of domain-specific entities, which can be distinguished based on their biological types. In this paper, we presented a system to identify biological entities with their types in a sentence. We showed that All Subsequence Entity Match against an automatically created ontology specific to the domain provides an efficient solution than Noun-based entity extraction. In addition, due to generic behavior of standard Entity extraction tools like Stanford-NER and Stanford-OpenIE, they fail to equate the level of performance achieved with ASEM-based system. Further-more, a high positive correlation between Recall obtained with ASEM-based system and Ontology size emphasizes that expansion of ontology can lead to a better system for this domain-specific knowledge (entity with its type) extraction task. Though we have shown the efficacy of our approach with the biomedical domain, we believe that it can be extended to any other domain where entities are domain specific and can be distinguished based on their types. For example, *financial domain, legal domain etc.*

References

1. Collins, M.: Ranking algorithms for named-entity extraction: Boosting and the voted perceptron. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics (2002) 489–496
2. Etzioni, O., Cafarella, M., Downey, D., Popescu, A.M., Shaked, T., Soderland, S., Weld, D.S., Yates, A.: Unsupervised named-entity extraction from the web: An experimental study. Artificial intelligence **165** (2005) 91–134
3. Baluja, S., Mittal, V.O., Sukthankar, R.: Applying machine learning for high-performance named-entity extraction. Computational Intelligence **16** (2000) 586–595
4. Carreras, X., Marquez, L., Padró, L.: Named entity extraction using adaboost. In: proceedings of the 6th conference on Natural language learning-Volume 20, Association for Computational Linguistics (2002) 1–4
5. Sudo, K., Sekine, S., Grishman, R.: Cross-lingual information extraction system evaluation. In: Proceedings of the 20th international Conference on Computational Linguistics, Association for Computational Linguistics (2004) 882
6. Asahara, M., Matsumoto, Y.: Japanese named entity extraction with redundant morphological analysis. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1, Association for Computational Linguistics (2003) 8–15
7. Laurent, D., Séguéla, P., Nègre, S.: Cross lingual question answering using qristal for clef 2006. In: Workshop of the Cross-Language Evaluation Forum for European Languages, Springer (2006) 339–350
8. Darwish, K.: Named entity recognition using cross-lingual resources: Arabic as an example. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Volume 1. (2013) 1558–1567
9. Daiber, J., Jakob, M., Hokamp, C., Mendes, P.N.: Improving efficiency and accuracy in multilingual entity extraction. In: Proceedings of the 9th International Conference on Semantic Systems, ACM (2013) 121–124
10. Yang, Z., Salakhutdinov, R., Cohen, W.: Multi-task cross-lingual sequence tagging from scratch. arXiv preprint arXiv:1603.06270 (2016)
11. Zhang, B., Lin, Y., Pan, X., Lu, D., May, J., Knight, K., Ji, H.: Elisa-edl: A cross-lingual entity extraction, linking and localization system. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations. (2018) 41–45
12. Cohen, W.W., Sarawagi, S.: Exploiting dictionaries in named entity extraction: combining semi-markov extraction processes and data integration methods. In: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM (2004) 89–98
13. Müller, H.M., Kenny, E.E., Sternberg, P.W.: Textpresso: an ontology-based information retrieval and extraction system for biological literature. PLoS biology **2** (2004) e309

14. Wang, W., Xiao, C., Lin, X., Zhang, C.: Efficient approximate entity extraction with edit distance constraints. In: Proceedings of the 2009 ACM SIGMOD International Conference on Management of data, ACM (2009) 759–770
15. Krallinger, M., Leitner, F., Rabal, O., Vazquez, M., Oyarzabal, J., Valencia, A.: Chemdner: The drugs and chemical names extraction challenge. Journal of cheminformatics **7** (2015) S1
16. Yimam, S.M., Biemann, C., Majnaric, L., Šabanović, Š., Holzinger, A.: An adaptive annotation approach for biomedical entity and relation recognition. Brain informatics **3** (2016) 157–168
17. Zheng, J.G., Howsmon, D., Zhang, B., Hahn, J., McGuinness, D., Hendler, J., Ji, H.: Entity linking for biomedical literature. BMC medical informatics and decision making **15** (2015) S4
18. Takeuchi, K., Collier, N.: Bio-medical entity extraction using support vector machines. Artificial Intelligence in Medicine **33** (2005) 125–137
19. Joseph, T., Saipradeep, V.G., Raghavan, G.S.V., Srinivasan, R., Rao, A., Kotte, S., Sivadasan, N.: Tpx: Biomedical literature search made easy. Bioinformation **8** (2012) 578
20. Sharma, R., Palshikar, G., Pawar, S.: An unsupervised approach for cause-effect relation extraction from biomedical text. In: International Conference on Applications of Natural Language to Information Systems, Springer (2018) 419–427