Unicage at DISTEMIST - Named Entity Recognition system using only Bash and Unicage tools

André Neves[1]

[1]*Unicage Europe, Lisbon, Portugal*

Abstract

This paper presents the participation of Unicage Europe in the DISTEMIST track, an international challenge focused on automatic detection of disease mentions in Spanish biomedical text using Natural Language Processing techniques. To tackle this challenge, a Named Entity Recognition (NER) dictionary based pipeline was developed using only Bash and Unicage commands. These commands allow the user to build highly efficient programs that can be combined in a modular way to build robust and flexible big data processing pipelines, even with limited hardware. With the goal of developing the first NER solution using Unicage technology, the DISTEMIST track was the right opportunity to apply it to the biomedical and multilingual domain. In the end, although the results were not exceptional, the developed system was capable of surpassing a competing system that also used a dictionary based approach.

Keywords

Unicage, Bash, Shell scripting, Named Entity Recognition

1. Introduction

Semantic indexing systems are important for many information retrieval tasks and are a crucial piece for many Natural Language Processing (NLP) solutions developed in the biomedical domain [1]. DISTEMIST (Disease Text Mining Shared Task) is the first track focusing on the automatic detection of disease mentions and their normalization in a dataset composed by 1.000 Spanish clinical case reports, with annotations linked to concepts from SNOMED-CT [2]. DISTEMIST is part of the 10th edition of BioASQ, a series of international challenges focused in promoting the development of state-of-the-art solutions for NLP tasks, namely semantic indexing and question-answering, for the biomedical domain.

To tackle the challenge proposed by DISTEMIST, the usage of the Unicage[1] commands and methodology was proposed to develop a simple and minimalist dictionary lookup Named Entity Recognition (NER) solution. Unicage offers a set of commands that allows the user to build efficient programs that can be combined in a modular way to build robust, yet flexible, big data processing pipelines. Unicage does not require expensive hardware or GPUs to work on, as it was designed to be efficient even with limited resources.

By participating in the DISTEMIST-entities subtrack, the goal was to successfully develop the first NER pipeline using Unicage technology and apply it to the biomedical and multilingual

CLEF 2022: Conference and Labs of the Evaluation Forum, September 5–8, 2022, Bologna, Italy

andre.neves@unicage.com (A. Neves)

[1]https://unicage.eu/

domain. Although the results achieved in the competition were not optimal, the system was successful in the task and was able to surpass another competing system. The system was also successful in being a minimalist solution, with every step of the pipeline being executed in a regular laptop with a virtual machine running Ubuntu, and using only OS built-in functions along with Unicage commands.

2. Background

2.1. Named-Entity Recognition

NER is a text mining task with the objective of recognizing mentions of relevant entities in text. It is essential for many NLP applications such as question answering, information retrieval, text summarization, among others [3], since it can locate entities of interest in the text that are related to the intended objective or domain of work.

There are several approaches for recognizing entities from text. The most common are based on Machine Learning techniques, such as Deep Learning and Transformer based architectures [4, 5]. These solutions require a corpus of text data labeled with the entities, along with their corresponding position in the text. The models are then trained using this corpus, so that they can learn to recognize the entities in new texts [6, 7]. However, not only they are dependent on potent hardware to be trained, but their efficiency is also strongly connected to the quality, quantity and availability of data to train the models, which can be a challenging in the multilingual biomedical domain [8, 9].

Another type of approach is the rule-based approach, which relies on manually defined rules and does not require annotated data. Among the rule-based approach is the dictionary lookup method, which uses a lexicon of specific terms or a knowledge base, such as an ontology, to match the text with the terms present in the lexicon. An example of these systems is MER (Minimal Named-Entity Recognizer) [10, 11], which presents a minimalist and flexible approach to easily identify terms in the text, only requiring the lexicon and the input text. The solution only uses built-in Unix shell commands and has minimal hardware requirements. MER is also capable of entity linking and it can be used with ontology data, thus providing additional information along with each identified term.

Finally, there are also hybrid approaches that combine Machine Learning techniques with rule-based systems [12, 13], which usually consists in a Machine Learning model that is later fine-tuned with additional linguistic rules to improve its evaluation metrics.

2.2. Unicage

As previously mentioned, Unicage consists in a shell scripting development methodology with focus on big data processing. Part of the Unicage methodology is a set of command-line tools. These tools are written in the C programming language and have been geared towards performance, leveraging the OS's memory and resource management capabilities to deliver high-speed processing. The variety of commands is also able to cover gaps that are present in the toolbox of built-in OS utilities, providing the user with additional tools for a plethora

of operations, from data manipulation and formatting, through complex math and statistical operations.

Additionally, Unicage follows the Unix philosophy [14], a methodology that emphasizes in building simple, short, clear, modular, and extensible code that can be easily maintained and repurposed. This paradigm ensures that systems can be customized and adapted to new needs according to the circumstances.

Unicage systems and data processing pipelines are composed of shell scripts that manipulate text files using a combination of Unicage commands and built-in OS utilities. The different stages of data processing inside the shell script are connected through pipes, an inter-process communication mechanism where the output of one command is connected to the input of the next command [15]. The pipes allow the OS to split each stage in several processes, which potentiates a better use of the system resources. In 2018, a study performed in the field of IoT showed that Unicage fared reasonably well when compared with other common big data technologies, surpassing them in some evaluation parameters in the context of batch and query processing [16].

A successful example of a data integration system using Unicage was developed for the LitCoin NLP Challenge[2], a 2-phase competition part of the NASA Tournament Lab, hosted by The National Center for Advancing Translational Sciences (NCATS), with contributions from the National Library of Medicine. The goal of the challenge was deploying data-driven technology solutions towards accelerating scientific research in medicine. In this challenge, Unicage Europe joined with a team of researchers from LASIGE[3], a research and development unit from the Faculty of Sciences of the University of Lisbon, in the fields of Computer Science and Engineering, to develop a state-of-the-art NLP solution. The joint solution reached the 7th place worldwide[4] and was the first successful integration of Unicage technology within a biomedical NLP pipeline.

3. Methods

3.1. Pipeline

Unicage Europe has decided to develop a simple dictionary lookup NER system made only with Bash and Unicage commands. This system shall be referred as Unicage-NER.

Unicage-NER is composed by a small pipeline of bash scripts, each one of them with a dedicated function, that can be seen in Figure 1. It receives as input the corpus and a lexicon file. The corpus is a text file where each record is composed by the identifier of the clinical case and the text of the clinical case from the DISTEMIST test set. The lexicon is a text file where each line is an entity of interest, sorted by alphabetical order. A more detailed explanation of the lexicons used can be seen in the next subsection.

The first script is dedicated to corpus normalization. First, the text is tokenized using Unicage's *maker* command, which can create records with N columns/words based on a key. The key will be the clinical case identifier and the *maker* command will transform each record so that there

[2]https://ncats.nih.gov/funding/challenges/litcoin

[3]https://www.lasige.pt/

[4]https://ncats.nih.gov/funding/challenges/litcoin/winners

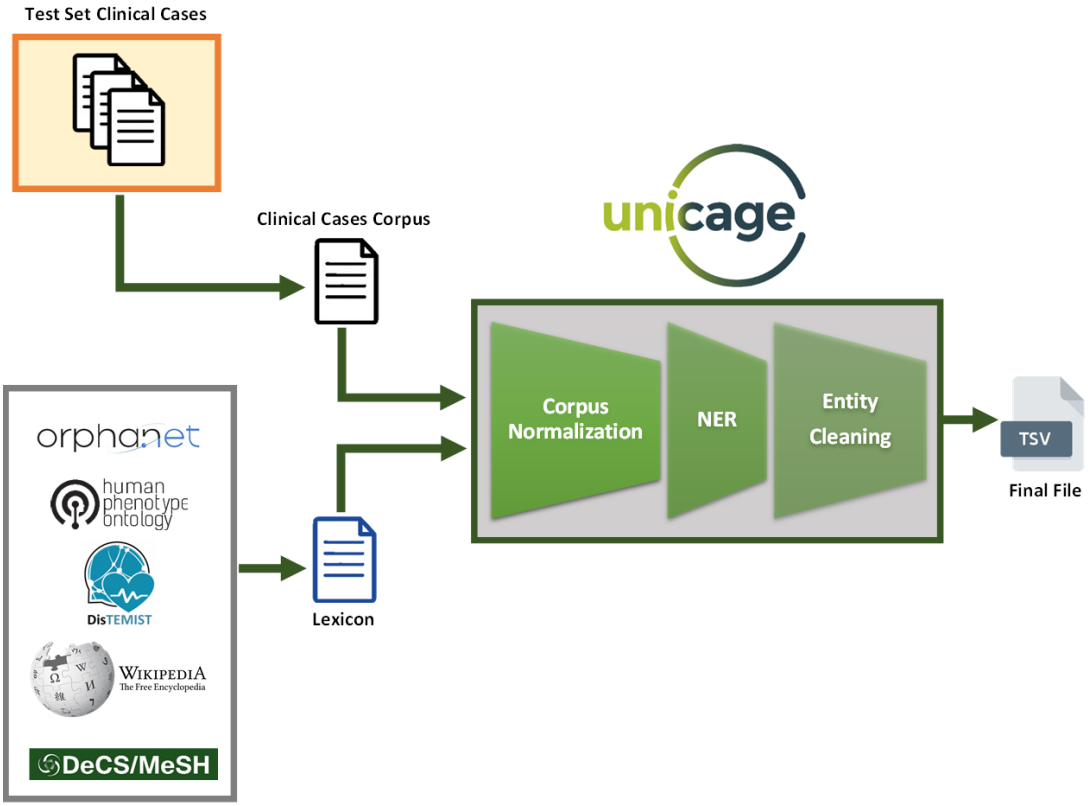

Figure 1: Unicage-NER script pipeline developed for the DISTEMIST-Entities subtrack. It receives as input the corpus composed by the clinical cases given by the challenge organization, and a lexicon file with the entities of interest. In the end, the found entities are converted to a tsv file, in the format required by the competition.

is a single word per line, instead of the whole text. Then, another Unicage command called *rank* is used to add the number of the word to each record. This step is necessary so that it is possible to keep the order of the words after future manipulation steps in the corpus.

Then, with the records organized in this way, it will be easier to calculate the starting and ending position of each word in the text. For this, Unicage's *uawk* is used. *uawk* is an optimized version of GNU awk [17], and is required since the found entities need to have their corresponding position in the text.

The final step of this script consists in converting all the characters to lowercase and removing all punctuation marks and special characters from the clinical cases. This is made using simple *sed* expressions and the built-in *tr* command.

Then, the NER script starts its execution receiving as input the lexicon and the output of the first script. The NER script works as follows: considering L as the maximum length of an entity in the lexicon, a total of L iterations are executed. For example, if the longest entity present in the lexicon is "miocarditis debida a covid-19", then L will be equal to 4, which corresponds to the number of words that comprise the entity. In each iteration, combinations of words, from 1 to

L, are made using Unicage's *maker* command. Then, these combinations of words are matched with the entities in the lexicon using Unicage's *join* command. *join* is a command that, given a sorted master file, which will be the lexicon, will retrieve from the corpus each entity that is also present in the lexicon, and stores them in a temporary file.

At this stage, an additional cycle is made where, in each iteration K, with K starting on 2 up to the value of L, the first N words of each clinical case are removed, where N = K − 1. This way, it is possible to analyze all combinations of consecutive words in the text of each clinical case, and extract entities that might not be detected in the first part of the process.

With all the entities identified, the final script receives the temporary file with the found entities and cleans them by removing all duplicates and overlapped entities using mainly a combination of *msort*, *uawk*, *self* and *delf* commands. *msort* is an Unicage command that applies a merge sort algorithm to the file based on a key field, whilst *self* and *delf* are two commands that allow an easy manipulation of the data fields on each record of the file. This way, it is possible to format the data before writing it in a tsv file with the format required by the challenge organization.

Unfortunately, due to time constrains, it was not possible to further complement this pipeline with the scripts necessary for the DISTEMIST-linking subtrack. However, it would be possible to create a Unicage solution that could link the found entities with the SNOMED-CT codes related to each of them.

3.2. Lexicons

For Unicage-NER to work, a lexicon with the entities of interest is required. For this process, lexicons comprising datasets with disease names, related terms and synonyms in Spanish were built. The lexicons can be divided in 2 major categories - XL and XXL.

The XL Lexicons are composed by:

- The entities present in the DISTEMIST training datasets for both sub-tasks, the cross-mappings files and the dictionary of terms present in the DISTEMIST Evaluation Library[5].
- DeCS (Health Sciences Descriptors)[6] terms and corresponding synonyms retrieved using the entities present in the cross-mappings files that had a DeCS code associated with it.
- HPO (Human Phenotype Ontology) [18] terms and corresponding synonyms retrieved using the entities present in the cross-mappings files that had an HPO code associated with it. The information retrieved from HPO was in English, so it had to be translated to Spanish. For this, the Google Translator API was used.
- List of rare diseases and corresponding synonyms retrieved from Orphanet's Orphadata[7] platform.
- A list of more than 1.000 diseases retrieved from the Spanish Wikipedia[8].

As to the XXL Lexicons, the same datasets that compose the XL lexicons were used in combination with the following:

[5]https://github.com/TeMU-BSC/DISTEMIST_evaluation_library
[6]https://decs.bvsalud.org/es/
[7]http://www.orphadata.org/cgi-bin/index.php
[8]https://es.wikipedia.org/wiki/Wikiproyecto:Enfermedades/Lista

- Entities present in the corpus of the CANTEMIST (CANcer TExt Mining Shared Task – tumor named entity recognition) subtask [19], a dataset composed by terms related to tumor morphology.
- A list of CIE10-Diagnósticos terms retrieved from the CodiEsp (Clinical Case Coding in Spanish Shared Task) subtask [20] present in the corresponding Github repository [9]. Only the entities present in the categories from A to N, together with the Q category, were extracted since these were the categories with terms most related with diseases.

Then, the same rules that were used to normalize the text described in the previous subsection were applied, i.e. every word was converted to lower case and all punctuation marks and special characters were removed. To identify the lexicons which only had this processing step, the suffix 'spc' was added.

Next, another variation of the lexicons was created by removing every term with 3 letters or less. This was decided since many of the terms with 3 letters or less letters corresponded to abbreviations that, due to the conversion to lower case, would be equal to other words with no relevant meaning in the medical domain, thus increasing the number of false positives. This filter was easily done using *uawk*. To identify the lexicons which had this processing step, the number '3' was added followed by the existing 'spc' suffix.

Another variation was tried, this one done with the usage of Python's NLTK's Snowball stemming algorithm[10] for the Spanish language. The algorithm was applied to the XXL Lexicon with the goal of normalizing the terms by reducing the words to their stem while removing the stop words. The same stemming algorithm was applied to the test set of the competition before giving it as input to Unicage-NER when this lexicon was used, so that the data had the same treatment as the lexicon. To identify this lexicon, the prefix 'STEM' was added to the lexicon name.

Finally, for all lexicons, the entries were sorted alphabetically and any duplicate entries that might be present were removed. A summary of the generated lexicons can be seen in Table 1 with the corresponding number of terms that comprise each lexicon.

Table 1
Summary of the lexicons used in the submissions made for the DISTEMIST-Entities subtrack.

Lexicon	Number of Terms
XL_Lexicon_spc	170.442
XXL_Lexicon_spc	195.243
XL_Lexicon_3spc	169.918
XXL_Lexicon_3spc	194.179
STEM_XXL_Lexicon_3spc	192.926

3.3. Hardware

To develop and run Unicage-NER, a simple laptop using a virtual machine was used. The virtual machine was running Ubuntu 20.04 LTS, with an 11th Gen Intel(R) Core(TM) i7-1165G7,

[9] https://github.com/TeMU-BSC/codiesp-evaluation-script/tree/master/codiesp_codes
[10] https://www.nltk.org/api/nltk.stem.snowball.html

2.80GHz, using only 4 cores, 4GB of RAM and a disk size of 115 GB.

4. Results

Unfortunately, due to time constrains, the Unicage team was only able to participate in the DISTEMIST-entities subtrack. In the end, the results were not optimal when compared with the other competing systems. The first results that were submitted had an error in the offsets of the entities that were identified, thus the score was virtually 0. However, the challenge organization allowed the submission of a corrected version of Unicage-NER system after the results were known. The results of Unicage-NER were compared with the best scoring system and the two systems from BSC (Barcelona Supercomputing Center) that were used as baseline, and can be seen in Table 2. The results achieved by Unicage-NER system are considered Post-Workshop.

Table 2
Results achieved by Unicage-NER in the DISTEMIST-entities subtrack compared with the best scoring system in the Micro-averaged F1 score (MiF1) and with the two systems submitted by the BSC which were used as baseline. The results from Unicage are considered Post-Workshop.
Measures: MiP – Micro-averaged Precision. MiR – Micro-averaged Recall. MiF1 – Micro-averaged F1 score.

Team Name	System Name	MiP	MiR	MiF1
PICUSLab	NER_Results	0.7915	0.7629	0.7770
BSC	DiseaseTagIt-Base	0.7146	0.6736	0.6935
Unicage	XL_LEX_3spc	0.2486	0.3303	0.2836
	XXL_LEX_3spc	0.2478	0.3306	0.2833
	STEM_XXL_LEX_3spc	0.2055	0.3464	0.2580
	XL_LEX_spc	0.2050	0.3380	0.2552
	XXL_LEX_spc	0.2045	0.3380	0.2548
BSC	DiseaseTagIt-VT	0.1568	0.4057	0.2262

5. Discussion

Compared with the best scoring system from PICUSLab and with DiseaseTagIt-Base from BSC, the scores achieved by Unicage-NER have a difference in MiP and MiF1 of more than 0.40. However, when compared with the DiseaseTagIt-VT from BSC, a dictionary lookup based on Levenshtein distance method, the results achieved by Unicage-NER were slightly better, with the exception in the MiR measure. The DiseaseTagIt-VT is a dictionary based system that uses the Levenshtein distance to look for the train and development entities in the test set. In contrast, Unicage-NER used lexicons comprising almost 200.000 possible terms, which might explain the slightly higher scores in MiP and, consequently MiF1 score.

Among the results achieved by Unicage-NER, it is also possible to notice that by stripping the lexicons of entities with 3 or less letters improved the scores, with an increase in MiP by almost 0.05 when compared with the lexicons where these entities were not removed. The only exception is the STEM_XXL_LEX_3spc lexicon that achieved a MiP score similar to the lexicons

with all the entities. However, the STEM_XXL_LEX_3spc was the lexicon that lead to the best MiR score by a slight margin when compared to the other lexicons.

Finally, it was expected that the submissions using the XXL Lexicons would achieve better overall results than the ones using the XL Lexicons, since the XXL Lexicons had more than 20.000 additional terms. However, the submissions using the XL Lexicons achieved slightly better scores than the ones that used the XXL Lexicons. This might be caused by the identification of terms that are not considered in the scope of the competition. For example, the dataset from CodiEsp is composed by a list of terms related to diagnostic of diseases and not exclusively to disease names.

6. Conclusion

Overall, the competition results showed that dictionary lookup methods alone are not optimal for these type of competitions. Nevertheless, Unicage Europe was able to successfully develop the first NER pipeline using Unicage technology with minimal infrastructure, and apply it to the biomedical and multilingual domain.

Unfortunately, it was not possible to complete the pipeline in time so that it could tackle the entity linking subtrack. However, as a future work, Unicage Europe intends to complete this pipeline and further improve its results, so that the solution could be more resilient on its own and, if possible, complement other existing NLP solutions. In addition, and following the best practices of benchmarking, Unicage Europe will continue to monitor the performance of future improvements and proprietary models against other NER solutions such as those presented in this competition.

Finally, Unicage Europe will continue its efforts in promoting scientific research and development in the areas of data processing and engineering, with the goal of developing efficient solutions with less amount of computational resources. The focus on the biomedical domain and in NLP area will remain a priority, since in these fields there is a plethora of data that is required to be efficiently processed in order to develop state-of-the-art solutions.

References

[1] A. Nentidis, G. Katsimpras, E. Vandorou, A. Krithara, L. Gasco, M. Krallinger, G. Paliouras, Overview of bioasq 2021: The ninth bioasq challenge on large-scale biomedical semantic indexing and question answering, in: K. S. Candan, B. Ionescu, L. Goeuriot, B. Larsen, H. Müller, A. Joly, M. Maistro, F. Piroi, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction, Springer International Publishing, Cham, 2021, pp. 239–263.

[2] A. Miranda-Escalada, L. Gascó, S. Lima-López, E. Farré-Maduell, D. Estrada, A. Nentidis, A. Krithara, G. Katsimpras, G. Paliouras, M. Krallinger, Overview of distemist at bioasq: Automatic detection and normalization of diseases from clinical texts: results, methods, evaluation and multilingual resources, in: Working Notes of Conference and Labs of the Evaluation (CLEF) Forum. CEUR Workshop Proceedings, 2022.

[3] J. Li, A. Sun, J. Han, C. Li, A Survey on Deep Learning for Named Entity Recognition, IEEE Transactions on Knowledge and Data Engineering (2020). doi:10.1109/tkde.2020.2981314. arXiv:1812.09449.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, volume 2017-Decem, 2017, pp. 5999–6009. URL: http://arxiv.org/abs/1706.03762. arXiv:1706.03762.

[5] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Brew, Huggingface's transformers: State-of-the-art natural language processing, ArXiv abs/1910.03771 (2019).

[6] P. Corbett, J. Boyle, Chemlistem: Chemical named entity recognition using recurrent neural networks, Journal of Cheminformatics (2018). doi:10.1186/s13321-018-0313-8.

[7] M. Krallinger, M. Pérez-Pérez, G. Pérez-Rodríguez, A. Blanco-Míguez, F. Fdez-Riverola, S. Capella-Gutiérrez, A. Lourenço, A. Valencia, The biocreative v.5 evaluation workshop: tasks, organization, sessions and topics, in: Proceedings of the BioCreative V.5 challenge evaluation workshop, 2017, pp. 8–10.

[8] A. Névéol, H. Dalianis, S. Velupillai, G. Savova, P. Zweigenbaum, Clinical natural language processing in languages other than english: Opportunities and challenges, Journal of biomedical semantics 9 (2018) 12. doi:10.1186/s13326-018-0179-8.

[9] E. Laparra, A. Mascio, S. Velupillai, T. Miller, A review of recent work in transfer learning and domain adaptation for natural language processing of electronic health records, Yearbook of medical informatics (2021). URL: http://hdl.handle.net/10150/662150. doi:10.1055/s-0041-1726522.

[10] F. M. Couto, L. F. Campos, A. Lamurias, MER: a Minimal Named-Entity Recognition Tagger and Annotation Server, Proceedings of the BioCreative V.5 Challenge Evaluation Workshop (2017).

[11] F. M. Couto, A. Lamurias, MER: A shell script and annotation server for minimal named entity recognition and linking, Journal of Cheminformatics (2018). doi:10.1186/s13321-018-0312-9.

[12] M. Fresko, B. Rosenfeld, R. Feldman, A hybrid approach to NER by MEMM and manual rules, in: International Conference on Information and Knowledge Management, Proceedings, 2005. doi:10.1145/1099554.1099667.

[13] R. Ramachandran, K. Arutchelvan, Named entity recognition on bio-medical literature documents using hybrid based approach, Journal of Ambient Intelligence and Humanized Computing (2021). doi:10.1007/s12652-021-03078-z.

[14] M. Gancarz, Linux and the Unix philosophy, Digital Press, 2003.

[15] A. Robbins, Linux programming by example: the fundamentals, Prentice-Hall, 2004.

[16] J. a. M. Moreira, H. Galhardas, M. L. Pardal, Leanbench: comparing software stacks for batch and query processing of IoT data, Procedia computer science 130 (2018) 448–455.

[17] A. V. Aho, B. W. Kernighan, P. J. Weinberger, The AWK programming language, Addison-Wesley Longman Publishing Co., Inc., 1987.

[18] S. Köhler, M. Gargano, N. Matentzoglu, L. C. Carmody, D. Lewis-Smith, N. A. Vasilevsky, D. Danis, G. Balagura, G. Baynam, A. M. Brower, T. J. Callahan, C. G. Chute, J. L. Est, P. D. Galer, S. Ganesan, M. Griese, M. Haimel, J. Pazmandi, M. Hanauer, N. L. Harris,

M. Hartnett, M. Hastreiter, F. Hauck, Y. He, T. Jeske, H. Kearney, G. Kindle, C. Klein, K. Knoflach, R. Krause, D. Lagorce, J. A. McMurry, J. A. Miller, M. Munoz-Torres, R. L. Peters, C. K. Rapp, A. M. Rath, S. A. Rind, A. Rosenberg, M. M. Segal, M. G. Seidel, D. Smedley, T. Talmy, Y. Thomas, S. A. Wiafe, J. Xian, Z. Yüksel, I. Helbig, C. J. Mungall, M. A. Haendel, P. N. Robinson, The Human Phenotype Ontology in 2021, Nucleic Acids Research 49 (2020) D1207–D1217. URL: https://academic.oup.com/nar/article-pdf/49/D1/D1207/35364524/gkaa1043.pdf. doi:10.1093/nar/gkaa1043.

[19] A. Miranda-Escalada, E. Farré, M. Krallinger, Named entity recognition, concept normalization and clinical coding: Overview of the cantemist track for cancer text mining in spanish, corpus, guidelines, methods and results, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2020), CEUR Workshop Proceedings, 2020.

[20] A. Miranda-Escalada, A. Gonzalez-Agirre, J. Armengol-Estapé, M. Krallinger, Overview of automatic clinical coding: annotations, guidelines, and solutions for non-english clinical cases at codiesp track of clef ehealth 2020, in: Working Notes of Conference and Labs of the Evaluation (CLEF) Forum. CEUR Workshop Proceedings, 2020.