Advances in Informatics, Management and Technology in Healthcare
J. Mantas et al. (Eds.)

doi:10.3233/SHTI220747

Mapping SNOMED CT Codes to Semi-Structured Texts via an NLP Pipeline

Sebastian KUNZ[a], Cyril ZGRAGGEN[a] and Murat SARIYAR[a,1]
[a]*Bern University of Applied Sciences, Switzerland*

Abstract. In the project presented here, we used NLP tools for annotating German medical trainings documents with SNOMED CT codes. Following research question was addressed: Is it possible to automate the annotation of training documents with an NLP pipeline especially designed for this task but requiring translation into English? The goal of our stakeholder, an institution responsible for the continuing education of physicians, was to facilitate the switch between different medical trainings programs by coding the same requirement with the same SNOMED CT code, even if the wording is different. We first describe how we chose the concrete NLP tools, after which the concrete steps for implementing our prototype are outlined: the NLP pipeline construction, the implementation, and the validation. We infer three important lessons from our results: (i) self-supervision is no free lunch and should be based on a sophisticated task, (ii) the translation via DeepL can be too context-dependent for a peculiar use case, and (iii) ontology extraction can increase efficiency as well as accuracy.

Keywords. NLP, word embeddings, spaCy, MedCAT, SNOMED CT

1. Introduction

Natural language processing (NLP) for clinical texts is now widely used for different tasks such as automatic summaries, text categorizations, or early adverse event detection [1]. In contrast to text mining, usually a reference corpus and a semantic understanding is required for solving these tasks. Clinical texts exhibit peculiarities that must be taken into consideration when applying NLP to them [2]. For example, different languages are used: Greek for describing diseases and symptoms, Latin for anatomical terms, and the mother tongue to communicate these terms and other information [3]. Hence, NLP tools developed for ordinary text, such as news text or Wikipedia entries, should most often be adapted to the medical context.

In the project presented here, we used NLP tools for annotating medical trainings documents with SNOMED CT codes. Following research question was addressed: Is it possible to automate the annotation of training documents with an NLP pipeline especially designed for this task but requiring translation into English? The goal of our stakeholder, an institution responsible for the continuing education of physicians, was to facilitate the switch between different medical trainings programs by coding the same requirement, e.g., with respect to clinical procedures, with the same SNOMED CT code in the medical training document, even if the wording is different. SNOMED CT is an

[1] Corresponding Author, Murat Sariyar, Bern University of Applied Sciences, Quellgasse 21, CH2502 Biel/Bienne, Switzerland; E-mail: murat.sariyar@bfh.ch.

ontology-based collection of medical terms that are defined and coded for an unambiguous use in clinical documentation and reporting.

In the following, we will first describe how we chose the concrete NLP tools, after which the concrete steps for implementing our prototype are outlined: the NLP pipeline construction, the implementation, and the validation. In the result section, the GUI, the NLP backend, and the validation results are presented. The discussion section describes the lessons learned as well as our next steps based on these insights.

2. Methods

As the project had only a time budget of 400-man hours, our first task was to choose an NLP framework, which could solve our task without too much initial training. For this task, we evaluated Python spaCy, Python NLTK and Apache cTake based on properties such as covering state-of-the-art NLP methods for the biomedical context, number of users, easy start, and supporting different kind of languages [4]. We rated each property with a score from 1 to 3 (low = 1, medium = 2 and high = 3), and summed all scores for each framework to reach our final decision. For the scoring, we relied on information from the scientific literature, blogs, tool documentation, and the experiences of the authors during the installation and familiarization with the NLP tools.

Constructing our NLP pipeline started with analyzing the medical training documents, which was important for understanding the structure, and to prepare the creation of a corpus. After that, we outlined the whole NLP process to be implemented. Four scrum sprints in two-weeks cycles were scheduled for the implementation. The validation was based on a gold standard, covering 133 SNOMED CT concepts, which was created by a manual annotation of five out of 45 training documents (our corpus), i.e., all relevant concepts in these documents were annotated by two staff members of our stakeholder. The F1 score was our main performance indicator.

3. Results

Our scoring of the NLP frameworks resulted in a clear conclusion: use spaCy, as it provides a large community, comprehensive documentation, a very easy start, and many tools adapted to the medical domain. As a promising tool for solving our task with our limited time budget, we decided to use the Medical Concept Annotation Toolkit (MedCAT [5]), which learns to extract medical concepts from unstructured (free) text and link them to a biomedical ontology such as SNOMED-CT. Its main features are (a) to combine named entity recognition with entity linking, (b) to contextualize concepts by training word embeddings (Word2Vec [6]) in a self-supervised way, and (c) to provide an initial model that can be used without further training. Especially the self-supervised training capacities seems innovative. In its self-supervision, it uses the unique names in SNOMED CT (one of the names assigned to each concept should be unique), find those names in the corpus, learns the embedding together with its context (choosing a certain window), and is then applied by comparing the resulting contextualized word embedding with the word embedding of new concept candidates [5].

The NLP pipeline based on MedCAT has the following components: (a) translating the documents from German to English using the DeepL API, (b) spell checking and cleaning of the text, tokenization, (c) lemmatization, and (d) finally the combination of

named entity recognition with entity linking. Before implementing the pipeline, two central components of MedCAT had to be determined, the vocabulary (VCB) and the concept database (CDB). VCB is a list of all possible words that can appear in the documents to be annotated and is primarily used for spell checking. We used the biomedical ScispaCy en_core_web_sm model for VCB. In CDB, each concept of SNOMED CT is represented by its ID and a name, and multiple names with the same meaning are mapped to the same ID (these are synonyms that are already captured by SNOMED CT). Steps (a)-(c) relied solely on Spacy, in step (d), we applied MedCAT, as described below. Hence, our innovation is related to the process integration of MedCAT via steps (a)-(d) and a GUI, as well as the validation of the whole pipeline via three scenarios.

For facilitating the application of the resulting NLP model, we implemented a GUI, named SNOMAST, using the PyQT5 framework that allowed using a GUI builder tool called Qt Designer. Figure 1 shows the SNOMAST tool with its different areas: importing a MedCAT model (1&2), reading in documents (3&4), entering the DeepL key for translations (5&6), displaying the imported or pasted text (7&8), producing as well as showing the mapping (9&10).

For the validation, we used three different scenarios: (i) untrained (SNOMED CT comparisons with some spelling checks), (ii) only self-supervised, and (iii) self-supervised as well as supervised learning with manually annotated data. Texts of three manually annotated training documents were used as the training set and the texts of the remaining two annotated documents acted as the test set. The results were disenchanting. The F1 measure values were 0.5 for all scenarios. The only difference between (i)+(ii) and (iii) was with respect to the recall/precision values: 0.43/0.6 against 0.46/0.55. Supervised learning had an additional value with respect to finding the complete concept "mediastinal lymph node", but instead of assigning to it 234263003 («excision of mediastinal lymph node»), it assigned the code 62683002 («mediastinal lymph node structure»). This motivates our lessons learned in the next section.

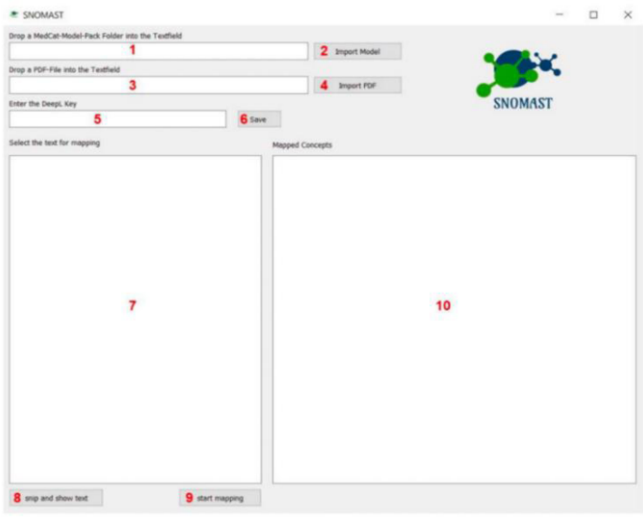

Figure 1. The implemented GUI, showing the different area for importing a MedCAT model (1&2), reading in documents (3&4), entering the DeepL key for translations (5&6), displaying the imported or pasted text (7&8), producing as well as showing the mapping (9&10).

4. Discussion

Coming back to our research question: translation from German to English via DeepL and the limited amount of training data for supervised learning seemed to prevent satisfying results for the automated annotation of training documents. We infer three important lessons from our results. First, word embeddings must be able to deal with peculiarities of a language, which can only be captured by high numbers of training data and a sophisticated self-supervised learning task. MedCAT tries to contextualize word embeddings, but the masked-language and next-sentence prediction tasks of BERT [7,8] together with its attention mechanism and the huge amount of data used in the self-supervision might be more promising in producing dynamic word embeddings.

Second, the translation of German into English via the DeepL API shows the other side of the coin with respect to word embeddings: the translation is more context-dependent than our use case requires. The authors witnessed such an issue with the translation of "Kaiserschnitt" as "Caeseran section" and "C-section" as well. Only in the former case, our NLP model recognized the concept. Either a more restricted translator or models such German-based BERT from huggingface should be used [9].

Third, allowing to map concept candidates to all SNOMED CT concepts proved to a major limitation of our work. On the hand, it slowed down our application significantly. On the other hand, it increases the probability of producing false positives. However, how to select the parts of SNOMED CT that are relevant? Here, ontology extraction methods based on the forgetting principle should be taken into consideration.

In conclusion, our results show that transfer learning for medical texts has certain pitfalls. The three lessons learned show our limitations and that we were not able to avoid pitfalls due to sub-optimal decisions. In an ongoing project, we are using BioBERT together with a restricted translator and an extracted set of SNOMED CT concepts for the classification layer. Preliminary results are promising, as they show much higher accuracy values as achieved than in the project described here. However, without the insights gathered in this project, such a boosting in accuracy would be improbable.

References

[1] Doan S, Conway M, Phuong TM, Ohno-Machado L. Natural language processing in biomedicine: a unified system architecture overview. Methods Mol Biol. 2014; 1168:275–294.
[2] Dalianis H. Clinical Text Mining: Secondary Use of EPRs. New York: Springer 2019.
[3] Pakhomov S, Pedersen T, Chute CG. Abbreviation and acronym disambiguation in clinical discourse. AMIA Annu Symp Proc. 2005; 589–593.
[4] Schmitt X et al. A Replicable Comparison Study of NER Software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate. In: 6th Int. Conf. on Social Networks Analysis. Management and Security. 2019; 338–343.
[5] Kraljevic Z et al. Multidomain clinical natural language processing with MedCAT: The Medical Concept Annotation Toolkit. Artif Intell Med. 2021; 117:102083.
[6] Jatnika D, Bijaksana MA, Suryani AA. Word2Vec Model Analysis for Semantic Similarities in English Words. Procedia Computer Science. 2019; 157:160–167.
[7] Devlin J et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics. 2019; 4171–4186.
[8] Lee J et al. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics. 2020; 36:1234–1240.
[9] Mihaela G et al.: Geolocating Swiss German Jodels Using Ensemble Learning. In: Proceedings of the Eighth Workshop on NLP for Similar Languages, Varieties and Dialects. Association for Computational Linguistics Kiev. 2021;84–95.