Humboldt Universität zu Berlin
Betreuer: Mario Sänger Mai 17, 2022

Exposé zur Masterarbeit

Entwicklung eines Korrekturmodells für BioNER

Jan Hoffschulte

1. Motivation

Biomedical Named Entity Recognition (BioNER) wird benutzt, um Entitäten aus biomedizinischen Texten zu extrahieren [1,2]. Bei den Entitäten kann es sich z.B. um Gene, Proteine oder Krankheiten handeln. Es existieren verschiedene Methoden, um BioNER zu realisieren [3]. Die meisten BioNER-Modelle benutzen Supervised Learning. Bei diesem Ansatz werden Modelle auf Daten mit bekannten Labels trainiert. Um eine hohe NER-Performance zu erreichen, werden qualitative annotierte Trainingsdaten benötigt. Häufig werden Domänenexperten wie z.B. Ärztinnen und Ärzte verschiedener Fachrichtungen bei dem Annotationsprozess eingesetzt, um einen qualitativen Goldstandard-Korpus zu erstellen. Diese Annotationsprozesse sind zeit- und kostenintensiv. Deswegen existieren oft nur kleine Goldstandard-Korpora, die für das Trainieren und Evaluieren von BioNER-Modellen benutzt werden können. Die geringen Datenmengen erschweren zudem die Generalisierbarkeit der BioNER-Modelle und damit die Anwendung auf ungesehene Texte [1,4].
Grundsätzlich wird versucht, den Erstellungsprozess von Trainingsdaten weitgehend zu automatisieren, um damit den manuellen Aufwand zu reduzieren. Ein semi-automatischer Ansatz ist beispielsweise zuerst mit einem BioNER-Modell einen Zielkorpus zu annotieren, der anschließend von Expertinnen und Experten korrigiert wird. Die Annotationsgenauigkeit von BioNER-Modellen ist häufig zu ungenau, sodass der Korrekturaufwand durch die Experten immer noch hoch ist [1]. Fehlende Generalisierbarkeit, Domänenwechsel oder andere Annotationsrichtlinien können die Vorhersagegenauigkeit der BioNER-Modelle verringern [1,5].
In dieser Masterarbeit wird untersucht, ob mit einem Korrekturmodell der manuelle Aufwand bei der Annotation von Korpora gesenkt werden kann. Für die Vorannotation eines Korpus kann ein BioNER-Modell benutzt werden. Dann wird ein Korrekturmodell eingesetzt, um die Annotationsgenauigkeit des vorannotierten Korpus zu verbessern. Mit Goldstandard-Korpora aus unterschiedlichen Domänen wird der Korrekturprozess simuliert. Es werden verschiedene Varianten des Korrekturmodells erprobt und evaluiert. Zudem wird analysiert, welchen Einfluss die Menge der Trainingsdaten auf die Güte des Modells hat.

2. Forschungsstand

NER ist ein weit untersuchtes Feld und aktuell dominieren transformer-basierte Modelle, die vortrainierte Sprachmodelle benutzen [3,6]. In der Regel wird ein BioNER-Modell auf einem Korpus trainiert und auf dem gleichen Korpus evaluiert. Es existieren eine Reihe von Arbeiten, die versuchen die Generalisierbarkeit von BioNER-Modellen mit unterschiedlichen Ansätzen zu verbessern [5]. Im Rahmen von Cross-corpus-Evaluationen werden BioNER-Modelle trainiert und auf einem ungesehenen Korpus anwendet, um die Generalisierung des Modells zu evaluieren [1,4,5]. Dadurch werden praxisorientiertere Szenarien getestet.

In Weber et al. [5] wird HunFlair, ein biomedizinischer NER-Tagger, in das Flair Framework integriert. HunFlair wird auf mehreren Korpora trainiert und lässt sich auf die Entitätstypen Zelllinien, Chemikalien, Krankheiten, Gene und Spezies anwenden. Es werden Cross-corpus-Experimente mit den drei ungesehenen Korpora CRAFT [7], BioNLP CG [8] und PDR [9] durchgeführt. Die F_1-Scores für die drei Korpora mit unterschiedlichen Entitätstypen liegen im Bereich zwischen 59-84%.

In Campos et al. [4] wird mit einem BioNER-Modell, basierend auf einem CRF-Klassifizierer, auf dem GENETAG-Korpus [10] trainiert und getestet. Es wird ein F_1-Score von 87,17% erreicht. Wenn dieser auf dem GENTAG-Korpus trainiert und auf dem CRAFT-Korpus mit den Entitätstypen Protein und Gen getestet wird, dann liegt der F_1-Score zwischen 45–55% [11]. Hier werden die Leistungsschwankungen bei der Cross-corpus Evaluation deutlich sichtbar. Ähnliche Ergebnisse wurden auch in Giorgi et al. [1] gefunden.

Nach bestem Wissen wurden keine Arbeiten gefunden, die automatisierte Korrekturverfahren für vorannotierte Korpora mittels Korrekturmodellen thematisieren. In Weber et al. [12] wird bereits ein ähnlicher Ansatz getestet, der mittels vorhergesagten Änderungsoperationen Graphen modifiziert. Mit einem Deep Learning-basierten Ansatz wird ein Sequence Labeling-Modell realisiert. Die für das Modell benötigten Embeddings werden mit dem Transformer-Model BERT erstellt. Die Modifikationen eines bestehenden Graphen werden textuell als Erweiterungen von Knoten formuliert. Für die Änderungsmöglichkeiten eines Graphen werden verschiedene Labels eingeführt, die von dem Modell vorhergesagt werden können. Diese Kernidee wird in dieser Arbeit aufgegriffen und auf die Korrektur von vorannotierten Korpora übertragen.

4. Konzeption

4.1 Methodisches Vorgehen

In der Abbildung 1 wird die allgemeine Vorgehensweise beschrieben, wie das Korrekturmodell trainiert und getestet wird. Zuerst wird ein BioNER-Modell basierend auf einem Goldstandard-Korpus bzw. auf mehreren Goldstandard-Korpora (A) trainiert. Dann wird das BioNER-Modell auf die Trainingsdaten von Korpus (B) angewendet. Basierend auf der Annotation der Trainingsdaten und der Goldstandard-Annotation des Korpus (B), werden die Trainingsdaten für das Korrekturmodell erstellt. Abbildung 1(a) veranschaulicht diese Pipeline. In der Abbildung 1(b) wird beschrieben, wie das Korrekturmodell auf die annotierten Testdaten von Korpus (B) angewendet wird, um Änderungsoperationen vorherzusagen.

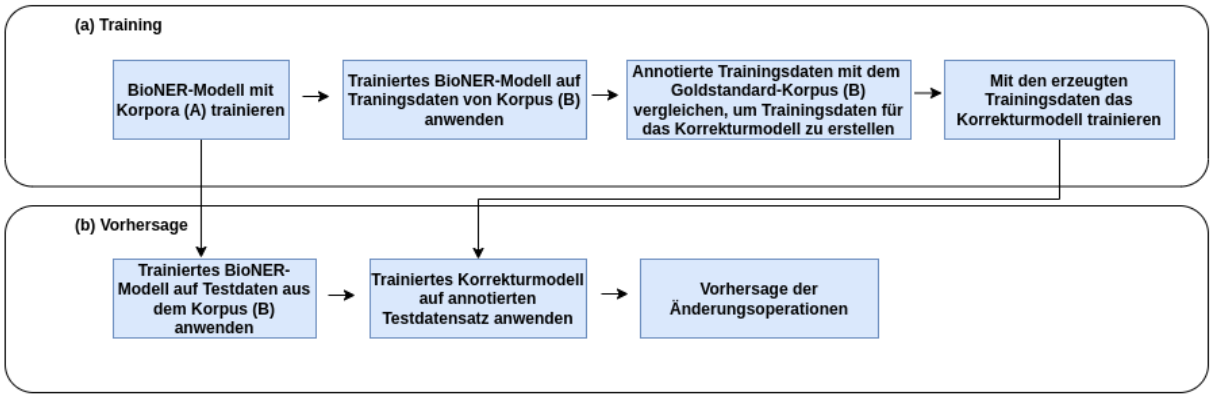

Abbildung 1: In Abbildung 1(a) wird beschrieben, wie das Korrekturmodell trainiert wird. Abbildung 1(b) beschreibt, wie Änderungsoperationen für einen Korpus vorhergesagt werden.

4.2 Datengenerierung für das Korrekturmodell

Mithilfe der in Abbildung 1(a) beschriebenen Pipeline werden die Trainingsdaten für das Korrekturmodell erstellt. In der Tabelle 1 wird anhand eines Beispiels beschrieben, wie die Trainingsdaten erzeugt werden. Hier wird exemplarisch der Entitätstyp Gen betrachtet. Mit einem BioNER-Modell werden die Wörter aus einem Satz des Korpus B gelabelt. Die Annotation wird mit dem entsprechenden Goldstandard-Korpus verglichen. Dadurch wird die erwartete Vorhersage der Änderungsoperationen ermittelt. Die Menge der Labels, die das Korrekturmodell verwendet, ergibt sich aus den möglichen Änderungsoperationen und dem verwendeten Tagging-Schema.

Satz	**Labeling durch ein NER-Modell**	**Goldstandard-Annotation**	**Erwartete Vorhersage des Korrekturmodells**
TP53	B-GENE	B-GENE	O
53	B-GENE	I-GENE	CHANGE-B-GENE-TO-I-GENE
und	O	O	O
BRAF	O	B-GENE	ADD-B-GENE
sind	O	O	O
Onkogene	B-GENE	O	DELETE-B-GENE

Tabelle 1: Beispiel für die Erstellung der Trainingsdaten anhand eines Satzes aus einem Korpus.

4.3 Modellarchitektur des Korrekturmodells

Das vorliegende Problem wird als Sequence Labeling modelliert. Abbildung 2 zeigt die Architektur des Korrekturmodells. Bei dem Korrekturmodell werden die Wörter aus einem Satz als Embeddings dargestellt. Die durch ein BioNER-Modell gelabelten Sequenzen aus der Vorannotation, werden als One-Hot-Embeddings repräsentiert. Diese werden dann mit den Embeddings aus dem Korrekturmodell konkateniert. Sie dienen als Eingabedaten für das Sequence Labeling-Modell, um Labels für jedes Wort vorherzusagen. Mit den vordefinierten Labels können die Änderungsoperationen für die Korrektur der Annotationen abgeleitet werden.
Die in 4.1 beschriebene BioNLP-Pipeline realisiert man mithilfe des Flair Frameworks [13]. Das Framework bietet verschiedene Korpora und Sprachmodelle an. Als Sprachmodell für das Korrekturmodell wird BioBERT verwendet. Das Sequence Labeling-Modell basiert auf einem Bidirectional Long Short-term Memory (BiLSTM) mit einem Conditional Random Field (CRF) Layer (BiLSTM-CRF). Die Vorhersage der Labels für die einzelnen Wörter erfolgt mit dem CRF-Layer. Das BiLSTM-CRF-Modell und das Transformer-basierte BioNER-Modell werden mit Flair erstellt.

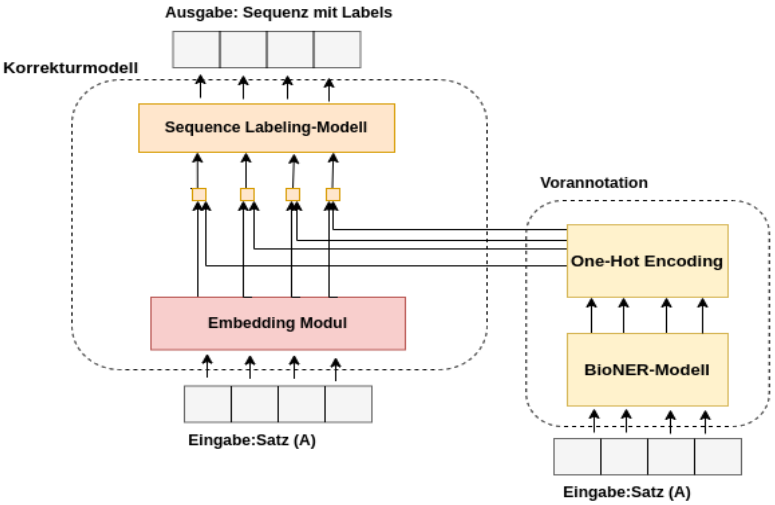

Abbildung 2: Architektur des Korrekturmodells. Realisiert werden die Modelle mit dem Framework Flair.

4.4 Evaluierung

Die Masterarbeit untersucht verschiedene Varianten des Korrekturmodells. Bei allen Varianten wird für die Erzeugung der Trainingsdaten für das Korrekturmodell zuerst ein in BioBERT-basiertes BioNER-Modell benutzt. Das Training des BioNER-Modells erfolgt auf einem oder mehreren Korpora (A). Dann werden mit dem trainierten BioNER-Modell die Daten eines anderen Korpus (B), die in diesem Fall als Trainingsdaten fungieren, annotiert. Anschließend werden, wie in 4.2 beschrieben, die Trainingsdaten für das Korrekturmodell erzeugt und das Modell trainiert. Bei den verschiedenen Testszenarien werden für das Korrekturmodell kontextualisierte Embeddings mithilfe des Transformers BioBERT erstellt oder Stacked Embeddings benutzt. Bei der Verwendung von Stacked Embeddings sind beispielsweise kontextualisierte Embeddings mit charakterbasierten Embeddings kombiniert.
Für die Evaluierung wendet man das trainierte BioNER-Modell auf den Testdatensatz des Korpus (B) an. Anschließend werden die Änderungsoperationen für den vorannotierten Datensatz mit dem Korrekturmodell vorhergesagt. Der Vergleich aller vorgestellten Varianten erfolgt mit einem BioBERT-basierten BioNER-Modell, das auf dem Korpus (B) trainiert und getestet wurde.
Die Testung erfolgt mit den Entitätstypen Gen und Krankheit aus verschiedenen Goldstandard-Korpora. In Tabelle 2 ist die Testkonfiguration für zwei Varianten beschrieben, bei dem das Korrekturmodell kontextualisierte Embeddings benutzt. Mit Variante 2 wird ein BioNER-Modell auf mehrere Korpora trainiert, um eine höhere NER-Performance zu erreichen [5].

Experiment	Training-BioNER	Training und Evaluation Korrekturmodell
1	CAFT [7]	BioNLP CG [8], FSU [14], DECA [15]
	NCBI Disease [16]	PDR [9], Scai Disease [17], BioCreative V CDR [18]
2	BioNLP CG, FSU, DECA	CAFT
	PDR, Scai Disease, BioSemantics	NCBI Disease

Tabelle 2: Verschiedene Testszenarien, um das Korrekturmodell zu evaluieren.

Weiterführende Analysen könnten den Vergleich weiterer BioNER-Tools einbeziehen. Dafür können die NER-Tools HunFlair [5], SciSpacy [19], tmChem [20] und Stanza [21] verwendet werden.

5. Literatur

[1] Giorgi, J. et al. (2020) Towards reliable named entity recognition in the biomedical domain. Bioinformatics, vol. 36, no. 1, pp. 280–286.

[2] Leser, U. et al. (2005) What Makes a Gene Name? Named Entity Recognition in the Biomedical Literature. Briefings in Bioinformatics, vol. 6, no. 4, pp. 357-369.

[3] He, Z. et al (2020) A Survey on Recent Advances in Sequence Labeling from Deep Learning Models. arXiv.org, https://arxiv.org/abs/2011.06727.

[4] Campos D. et al. (2013a) Gimli: open source and high-performance biomedical name recognition. BMC Bioinformatics, 14, 54.

[5] Weber, L. et al. (2021) HunFlair: An Easy-to-Use Tool for State-of-the-Art Biomedical Named Entity Recognition. Bioinformatics, vol. 37, no. 17, pp. 2792–2794.

[6] Huang, M. et al. (2020) Biomedical named entity recognition and linking datasets: survey and our recent development. Briefings in Bioinformatics, vol. 21, no. 6, pp. 2219–2238.

[7] Bada, M. et al. (2012) Concept annotation in the craft corpus. BMC Bioinformatics, 13, 161.

[8] Pyysalo, S. et al. (2013) Overview of the cancer genetics (CG) task of BioNLP shared task 2013. In: BioNLP Shared Task 2013 Workshop. Association for Computational Linguistics, Sofia, Bulgaria, pp. 58–66.

[9] Kim,B. et al. (2019) A corpus of plant–disease relations in the biomedical domain. PLoS One, 14, e0221582.

[10] Tanabe, L. et. al. (2005) GENETAG: a tagged corpus for gene/protein named entity recognition. BMC Bioinformatics, 6 Suppl 1(Suppl 1):S3.

[11] Campos D. et al. (2013b) A modular framework for biomedical concept recognition. BMC Bioinformatics, 14, 281.

[12] Weber L. et al. (2021) Extend, don't rebuild: Phrasing conditional graph modification as autoregressive sequence labeling. Association for Computational Linguistics. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 1213–1224.

[13] Akbik, A. et al. (2019) FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP. Association for Computational Linguistics, In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations), pp. 54–59.

[14] Hahn, U. et al. (2010) A Proposal for a Configurable Silver Standard. Association for Computational Linguistics, In Proceedings of the Fourth Linguistic Annotation Workshop, Uppsala, Sweden, pp. 235–242.

[15] Wang, X. et al. (2010) Disambiguating the species of biomedical named entities using natural language parsers. Bioinformatics, vol. 26, no. 5, pp. 661–667.

[16] Doğan, R. et al. (2014) NCBI disease corpus: A resource for disease name recognition and concept normalization. Journal of Biomedical Informatics, vol.47, pp. 1-10.

[17] Gurulingappa, H. et al. (2010) An Empirical Evaluation of Resources for the Identification of Diseases and Adverse Effects in Biomedical Literature, 2nd Workshop on Building and evaluating resources for biomedical text mining (7th edition of the Language Resources and Evaluation Conference).

[18] Li, J. et al. (2018) BioCreative V CDR task corpus: a resource for chemical disease relation extraction. Database (Oxford), vol. 2016, article ID baw068.

[19] Neumann, M. et al. (2019) ScispaCy: fast and robust models for biomedical natural language processing. In: 18th BioNLP Workshop and Shared Task. Association for Computational Linguistics, Florence, Italy, pp. 58–66.

[20] Leaman, R. et al. (2015) tmchem: a high performance approach for chemical named entity recognition and normalization. J. Cheminf., 7, S3.

[21] Zhang, Y. et al. (2021) Biomedical and Clinical English Model Packages in the Stanza Python NLP Library. Journal of the American Medical Informatics Association, vol. 28, no. 9, pp. 1892–1899.