

Transfer Learning Methoden für Named Entity Recognition in Benutzer-generierten Texten

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Data Science

eingereicht von

Johannes Bogensperger, Bsc.

Matrikelnummer 01427678

an der Fakultät für Informatik
der Technischen Universität Wien

Betreuung: Univ. Prof. Dr. Allan Hanbury
Mitwirkung: Projektass. PhD Gábor Recski
Dr. Sven Schlarb

Wien, 31. Mai 2021

Johannes Bogensperger

Allan Hanbury

Exploring Transfer Learning Techniques for Named Entity Recognition in Noisy User-generated Text

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Data Science

by

Johannes Bogensperger, Bsc.

Registration Number 01427678

to the Faculty of Informatics

at the TU Wien

Advisor: Univ. Prof. Dr. Allan Hanbury

Assistance: Projektass. PhD Gábor Recski

Dr. Sven Schlarb

Vienna, 31st May, 2021

Johannes Bogensperger

Allan Hanbury

Erklärung zur Verfassung der Arbeit

Johannes Bogensperger, Bsc.

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 31. Mai 2021

Johannes Bogensperger

Danksagung

Diese Diplomarbeit basiert auf Forschungsarbeiten, die im Rahmen des von der EU geförderten COPKIT-Projekts durchgeführt wurden, das im Rahmen des Forschungs- und Innovationsprogramms Horizon 2020 der Europäischen Union unter der Fördervereinbarung Nr. 786687 gefördert wurde.

Hiermit möchte ich meinem Betreuer Gábor Recski meinen aufrichtigen Dank für die Betreuung dieser Arbeit und für sein kontinuierliches Feedback und seine Tipps während des gesamten Forschungsprojekts aussprechen. Des Weiteren möchte ich mich bei meinem Betreuer am AIT Dr. Sven Schlarb und dem Austrian Institute of Technology für die Möglichkeit der Teilnahme an diesem Forschungsprojekt und deren Unterstützung bedanken. Mein Dank geht auch an meine Eltern und den österreichischen Staat für die Finanzierung meiner Ausbildung.

Acknowledgements

This article is based on research undertaken in the context of the EU-funded COPKIT project, which has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 786687.

I would like to express my sincere gratitude to Gábor Recski for supervising this thesis and for his continuous feedback and guidance throughout the research project. Furthermore, I would like to extend my sincere thanks to my AIT supervisor Dr. Sven Schlarb, and the Austrian Institute of Technology for the opportunity to participate in this research project and their support. My appreciation also goes to my parents and the Austrian state for funding my education.

Kurzfassung

Strafverfolgungsbehörden sind interessiert, aktuelle Trends und Entwicklungen in Darknet-Märkten zu erkennen. Das Extrahieren von Informationen für solche Märkte erfordert Wissen über die enthaltenen Entitäten, welches über Named Entity Recognition (NER) extrahiert werden kann. Moderne NER-Modelle werden mittels Supervised Learning optimiert, aber annotierte Datensätze für spezifische Anwendungsdomänen, wie Drogenerkennung in Darknetmärkten, sind kaum vorhanden. In dieser Arbeit haben wir einen NER-Datensatz erstellt, welcher sich auf Drogen in Darknet-Märkten konzentriert, und Ressourcen und Techniken zur Domänen- und Aufgabenanpassung evaluiert. Der Datensatz wurde mittels Crowd-Sourcing erstellt und ist etwa viermal so groß wie der einzige andere derzeit verfügbare NER-Datensatz für Darknet-Märkte. Im Zuge der Arbeit stellten wir fest, dass wir unsere NER-Vorhersageleistung durch Domänenanpassung verbessern konnten, indem wir unsere Sprachmodelle auf Darknet-Texten und reduzierten Versionen von Wikipedia-Texten über illegale Drogen feinabgestimmt haben. Unser Modell war in der Lage, Drogenentitäten mit einem F1-Score von bis zu 84.04 Punkten nach der CoNLL2003 NER-Evaluationsmetrik vorherzusagen.

Abstract

Modern law enforcement agencies strive to identify current trends and developments in Darknet markets. Extracting information from such markets requires knowledge about the contained entities, which can be extracted via Named Entity Recognition (NER). Modern NER models are trained via supervised learning, which requires an annotated dataset, but such datasets for specific application domains, e.g. drug detection in Darknet markets, are rarely available. In this work, we created a NER dataset focused on drugs in Darknet markets and evaluated resources and techniques for domain and task adaptation of our NER models. The dataset, with about 3.500 item listings, was created via crowd-Sourcing and refined via a manual review. It is approximately four times the size of the only other available NER dataset for Darknet markets, we were aware of at this time. We found that we were able to improve our NER prediction performance by domain adaptation via fine-tuning our language models on Darknet item descriptions and reduced versions of Wikipedia texts about illicit drugs. Our models were able to predict drug entities with a F1-Score of up to 84.04 points according to the CoNLL2003 NER evaluation metric.

Contents

Kurzfassung	xi
Abstract	xiii
Contents	xv
1 Introduction	1
1.1 Structural Overview	1
1.2 Motivation	1
1.3 Our Approach and Main Results	3
2 Related Work / Literature Review	7
2.1 Named Entity Recognition	7
2.2 Language Modelling / Transfer Learning	9
2.3 Named Entity Recognition in Noisy User-generated Texts	10
2.4 Crowd-Sourcing	11
3 Approach	13
3.1 Datasets	13
3.2 Crowd-sourcing	14
3.3 Final Dataset Statistics	21
3.4 Other Datasets	24
3.5 Model Architecture	25
3.6 Name Entity Recognition	28
4 Experimental Evaluation	31
4.1 Experiment Design	31
4.2 Evaluation Metrics	32
4.3 Results - General Performance Evaluation	33
4.4 Results - Task Adaptation	35
4.5 Prediction Pattern Evaluation	37
5 Conclusion and Future Work	41
5.1 Research Questions	41
	xv

5.2 General Results	43
Bibliography	45
A Appendix	51
A.1 Project Execution - Detailed	51
A.2 Annotation Guidelines	59
List of Figures	71
List of Tables	73

Introduction

1.1 Structural Overview

Chapter 1 explains the motivation behind this thesis and provides a brief overview of the Named Entity Recognition (NER) task, conducted work and achieved results. In chapter 2 we are going to provide an overview of related literature in the field of NER, reference the Transformer based language models and related work on NER in noisy user-generated texts. In chapter 3 we are going to describe the dataset creation process and our model architecture. Chapter 4 describes the experiments conducted in the scope of this study and subsequently their results. Chapter 5 elaborates on which conclusions to draw from our results and possible future work. Details of the dataset creation process and specific annotation guidelines can be found in the Appendix in section A.1 and A.2.

1.2 Motivation

Nowadays the vast majority of data in the internet is present in an unstructured form. In order to leverage this data, scientists developed various Natural Language Processing (NLP) methods to extract information from unstructured data sources. One of the most fundamental techniques is called Named Entity Recognition (NER). NER strives to identify entities of specific types, such as Persons, Organizations or Locations, in text corpora. According to [GGK18] the definition of Named Entities is: "A named entity is a word form that recognizes the elements having similar properties from a collection of elements.". The task of NER was introduced at the 6th Message Understanding Conference in 1996 [GS96] and remains a relevant challenge for researchers since then. A variety of modern NLP techniques use NER results as input for further analysis. Possible examples for NER in the domain of medicine could be diseases as entity type or minerals in the geology domain. This master thesis focuses on Darknet markets so possible entities

could be items for sale such as drugs, guns or virtual/digital goods for fraud (e.g. credit card data).

Since 1996 the techniques used to conduct NER evolved substantially. State-of-the-art (SOTA) NER Models achieve a good performance on common text corpora such as newswire or scientific text. [GGK18] Unfortunately, the performance of these techniques rapidly decreases once they are applied to domain specific texts, uncommon entity types or especially in noisy user generated text such as tweets. [DNEL17] In 2017 the Workshop of Noisy User-Generated Texts published a NER challenge upon texts from various internet sources (e.g. Reddit or Twitter). The best papers achieved a significantly worse performance compared to NER systems used for common text, which indicates that this problem is not being solved yet.

We assume that this is also because of the lack of resources, also called data scarcity. Except for Twitter datasets such as [DBR16] there aren't a lot of datasets available for noisy user-generated data, especially when dealing with specific use-cases such as recognizing uncommon entity types.

This master thesis is based on the requirements of international and local law enforcement agencies to detect illegal offerings on Darknet Markets (DNMs). It is embedded alongside other related research within the scope of project Copkit (<https://copkit.eu/>). This project is focused on creating intelligence-led Early Warning and Early Action Systems for European law enforcement agencies. Extracting information about current offerings on Darknet Markets such as guns, drugs or fraud services can provide valuable insight on current developments of illicit activities.

Since those user-generated texts on such platforms do not follow common grammatical rules, lack proper punctuation and use a lot of slang words many NLP models cannot perform their tasks in a satisfying manner. As mentioned before, there is very little annotated data available in the domain of Darknet markets, since organizations which are concerned with such security relevant tasks are unlikely to share their data. We found only one dataset from Al-Nabki [NFM19], which can be considered quite small with 851 samples. Moreover, the entities in focus are general products without specific typing for drugs, guns or others. Therefore, the main contribution of this thesis is a new dataset for the recognition of drug entities in Darknet Market item descriptions and baseline NER systems built upon the dataset. For these baseline systems we are going to evaluate NER model architectures and performance improvement measures for such a noisy dataset. This dataset is only available via contacting the author by email¹, whilst providing information about your research interests and institutional affiliation. We hope that the dataset will catalyse further research on this topic.

1.2.1 Task Description

International police organizations are interested in detecting illegal trading activities in Darknet markets. In our use case we are going to investigate how we can detect and

¹DreamDrugDataset@gmail.com

extract (mostly illicit) drug entities from item listings of Darknet markets. Therefore, we annotated data from Darknet markets and created NER models to extract text spans which contain named entities of the type "drug". The exact definition of entities, which shall be extracted can be found in section 3.2.1. Since, there is little to no annotated data available for this task, we created a new dataset for illicit Drug NER, which is available for further research.

Some might consider the NER task as solved, since for common domains and entity types current models achieve a good performance. This performance drops heavily once we move away from newswire text or common entity types like Person or Organization. Major challenges remain unsolved until today due to the ambiguous structure of language, its wide variety and fast development.

The main challenges for the NER task following [GGK18] are our domain, genre and the uncommon entity type (illicit) drug. The most important characteristics of our texts stem from the mixture of internet slang and illicit drug user slang. They often refer to certain advertised goods in a way, which is not intuitive to the user. E.g. "Green Hulks" are Xanax tablets or "Blue Mitsubishis" are Ecstasy tablets. Furthermore, we assume that the following challenges are relevant for our work:

- **Annotation of training data:** Supervised learning methods are based on labelled training data. This study focuses on supervised learning models making the existence of training data a prerequisite. The quality of our training data will define the upper boundary of the models performance. If the training data is of bad quality the model will be prone to errors.
- **Lack of resources:** Our Darknet market domain, equally to many other specialized domains, suffers heavily from the lack of annotated datasets. If a lot of training data is available, models tend to perform better.
- **Ambiguity in text:** Text tokens can be ambiguous. For Example, "Speed" can refer to the measure of the rate of motion, a chocolate bar or the drug "Amphetamine". Models for Natural Language Processing tasks needs to disambiguate text tokens based on their context.
- **Nested entities:** Entities can contain other sub-entities. An example in the Darknet domain could be that the token of type "drug" (e.g. "Acetaminophen M367") contains a unique descriptor of the drug, but also a pill imprint. With neural network based approaches, it is hard to force the model to learn an existing type hierarchy, so the model understands that these two concepts will not oppose each other.

1.3 Our Approach and Main Results

In this subsection we provide a brief overview on where our approach stems from and what the main results are. Detailed information upon the theoretical background can be found in section 2.1 and the results are explained in more detail in section 5.

Based on previous work for cross-domain NER (see section 2.1), we strive to build a NER model trained on common datasets, which can be adapted to other, more specific, NER tasks. This means that we use an extensive NER dataset for learning the NER task in the general domain and fine-tune our model on the smaller dataset of the target domain. Existing approaches build upon standard corpora from newswire, Wikipedia or scientific text, which perform poorly on out-of-domain texts. [LXY⁺20] For example, the original [LXY⁺20] paper trained their model initially on the ConLL2003 [TKSDM03] dataset which consists of newswire texts. In contrast to previous approaches we will build an NER system based on noisy user-generated text corpora such as the W-NUT 2017 corpus [DNEL17] or the Broad Twitter Corpus [DBR16]. We assumed that this will increase the prediction performance of our model in the noisy target domain of Darknet markets. In our model architecture we leveraged a Transformer architecture [DCLT18, VSP⁺17] for word embeddings.

Earlier approaches such as [JXZ19] used CBS SciTech News as target domain which was quite similar to the original newswire corpus. Other approaches which focused on noisy text from Twitter [LNC⁺18] did not focus on specific topics or domains. They rather tried to recognize general entity types like Persons or Organizations within the widespread range of topics covered by tweets. [LXY⁺20] focused on specialized domains and entity types, in opposition to [LNC⁺18]. Moreover, set-up of [LXY⁺20] contained less overlap between source and target domain in terms of entity types and vocabulary, but still achieved a performance boost by pre-training on a NER dataset with only a small set of common entity types (Person, Organization, Location, Misc). In this thesis we evaluated if this pre-training would increase the prediction performance in case of our NER task for drug detection as well.

Our main contributions can be summarized as:

- A drug NER dataset, called DreamDrug, from Darknet markets: This dataset consists of over 3.500 item listings with over 360.000 words and covers mostly illicit drugs annotated by crowd-workers. Furthermore, those annotations were manually corrected according to our annotation guidelines (see section 3.2.1 or A.2). The final dataset is of gold standard quality according to the aforementioned definitions. Our crowd-workers reached a fair up to substantial agreement on the annotations (up to 0.76 in terms Cohen’s Kappa - see Table 3.1 and Figure A.1) and it is roughly four times the size of the biggest comparable dataset currently available [NFM19], according to our knowledge.
- We explored the most suitable options for domain adaption transfer learning (see section 2.1), also known as language model fine-tuning, for Darknet markets.
- We evaluated possible performance improvements via task adaptation transfer learning (see section 2.1).

Based on the the previous work done on transfer-learning options by [LXY⁺20, YK20a] we were curious how those findings would translate or if they even apply in domains with noisy text structure, misspellings, code-words, ASCII art, slang words and no proper grammatical rules or punctuation. Therefore the following research questions emerged:

1. When using pre-trained models (details of pre-training / task adaption are described in section 2.1) for our NER task, will they achieve a higher performance in terms of F1-score, if the textual structure of the source domain is similiar to the text structure of the target domain?

I.e. will noisy user generated source domains (e.g. internet datasets as BTC or W-NUT 2017 - see section 3.1.1) be a better starting point for Darknet NER than classic NER corpora such as newswire corpora ConLL2003 [TKSDM03]?

2. Can pre-training on well-structured text corpora increase the F1-score of our NER model, even though the target domain is different?

I.e. Will pre-training on Wikipedia illicit drug text benefit the final model even though its grammatical structure is quite different? Or is the only well suited text corpus for pre-training directly selected from Darknet markets?

3. Are distantly supervised datasets such as Wikipedia articles aligned with a Knowledge Graph (see section 3.4.1) able to further boost the F1-score of a NER system with specialized vocabulary and noisy texts?

4. Do our custom NER models perform better, in terms of F1-score, compared to off-the-shelf models such as the one from Akbik et. Al 2019 [ABV19, ABB⁺19] also known as FLAIR?

In our experiments for research question 1 we didn't find any evidence to support the claim that the text structure will have an positive impact on the performance of the transfer learning approach (task adaption). We couldn't gain a significant performance increase by pre-training our models on other NER corpora (see results in section 4.4). This is the reason why we reject the hypothesis that text structure similarity impacts task adaption performance (see section 2.1) in a positive way.

We found that domain adaptive pre-training (domain adaptation - see section 2.1) using additional well structured texts from Wikipedia about illicit drugs (see section 3.4.2) helped to further increase the F1-performance of our models. This is sufficient evidence to accept the hypothesis from research question 2, that LM fine-tuning on well structured texts about domain relevant topics can increase the prediction performance. The results we use as evidence can be found in section 4.3.

We were not able to further improve the prediction performance in terms of F1-Score by pre-training our models on our distantly supervised dataset (see results in section 4.4). It rather decreased the prediction performance and therefore we cannot provide

evidence for this hypothesis. However, there might be more appropriate resources for distant supervision. E.g. a specialized drug ontology such as the one from [Sha17] could probably cover more prescription drugs and would enable a more accurate entity linkage between text and Knowledge Graph. Therefore, research question number 3 is left open for further research.

Finally, our models outperformed our example for an off-the-shelf model (FLAIR [ABV19]) trained on our dataset by 10-12 points in terms of F1-Score. Therefore, we can state that research question 4 was answered positively and we were able to achieve a competitive edge by our custom models (see Table 4.5 and 4.6).

Related Work / Literature Review

2.1 Named Entity Recognition

NER models such as [HMLB20, FGM05, ABB⁺19] are typically focused on extracting a restricted set of entities from text corpora such as newswire (e.g. CoNLL2003 - [TKSDM03]), OntoNotes - [HMP⁺06]) or Wikipedia texts (WikiNER - [NRR⁺13]). Typical entity types for NER in general domains are Organization, Person or Location [TKSDM03, HMP⁺06, NRR⁺13].

Since the introduction of the task in 1996, a variety of techniques were used to extract Named Entities. Early approaches build upon handcrafted rule-based algorithms, which provided good results for specific domains [GGK18]. Modern systems rely on machine learning based algorithms to overcome the weaknesses of rule based systems, which often lack generalizability and require high efforts and skill to build and maintain such models. Currently neural models based on Convolutional Neural Network (CNN) approaches such as [AMLMS17, NFAFR20] are replaced by architectures using Transformers [DCLT18] for word embeddings with a fully connected layer or a Conditional Random Field model (CRF) as classification layer such as [JXZ19, JZ20, LXY⁺20, YK20a].

The so called Conditional Random Field architecture [LMP01] is a graph-based discriminative model architecture, which is well suited for prediction tasks where the predicted label (state) is dependent on the prediction of neighbouring inputs. Therefore, CRFs are often used in NLP tasks, such as Part-Of-Speech tagging or NER. In those cases the predicted label of a token, is dependent on the prediction of its context. This architecture overcame the biggest drawback of Maximum Entropy Markov Models (MEMMs - [MFP00]), which suffered bias towards states (labels) with few successor states.

This master thesis builds upon the CrossNER architecture [LXY⁺20], which is focused on cross-domain NER. This architecture (and cross-domain NER in general) is based upon the hypothesis that learning the NER task previously on a generic dataset from a source domain will increase the performance of the model in a target domain. We will refer to this transfer learning technique as **task adaptation**, in contrast to **domain adaptation** or domain adaptive pre-training of the language model, which is used to fine-tune the word-embeddings.

For most use-cases where NER is applied in specialized domains the vocabulary and entity types are not properly represented in popular NER training corpora. Earlier cross-domain NER training corpora were based on newswire articles, scientific papers or Wikipedia and focus on detecting common entities like Persons, Organizations and Locations. Unfortunately, those are often not relevant in practice where more fine grained entity types are required to be extracted from alternative text genres and domains. Even well trained NER models fail to generalize to these different domains due to domain discrepancy. [LXY⁺20]

Creating extensive high quality text corpora for individual applications is not feasible in most situations due to time and cost constraints. These scarcity issues shall be compensated (at least partially) by learning the NER task in advance in a different domain. The CrossNER paper [LXY⁺20] created their own NER datasets for multiple domains, since other cross domain NER studies usually benchmark their results on similar domains with little domain discrepancy or with strongly overlapping entity types. The authors in [LXY⁺20] found that this does not properly reflect the conditions found in practice and therefore created their own datasets to better represent common use-cases.

The datasets created in [LXY⁺20] cover five specialized domains, namely Artificial Intelligence, Music, Literature, Natural Science and Politics. They created 1000 samples for each domain where only 100-200 are used as a training dataset and the remaining data is used to accurately measure the performance of the final NER model. This set-up with 100 training examples captures the setting of real life applications, since it should not impose a problem to create a training dataset of this size. We will refer to a setting with only 100 examples as **Few-Shot** setting in this paper. They found that, despite its simple design, their model outperformed state-of-the-art models, when trained on a source domain and afterwards on the target domain (including domain adaptive pre-training to adjust to target domain vocabulary). Unlike other state-of-the-art models [JXZ19, JZ20] their models did not use a Conditional Random Fields model. Instead they used a simple linear layer on top of the language model.

Few-Shot learning is a popular set-up in current research in general [SSZ17] as well as in NLP tasks such as NER [HLS⁺20, HKGNH18]. Traditional NER models are built upon extensive annotated text corpora in a supervised learning set-up. The reason for the popularity of Few-Shot learning are the same data scarcity issues, already mentioned at cross-domain NER. Building text corpora is resource intensive, requires expert knowledge of the domain and a high amount of time and budget. This often prevents the usage

of NER models in real-world applications. In practice often only a few examples are available for training models.

Recent works created various methods to overcome the need for extensive training corpora and work with a Few-Shot set-up. Prototypical Networks from [SSZ17] are an architecture which tries to solve this issue. They have already been applied on the NER task by [FLK19]. However, this is not the only approach where Few-Shot learning is used. [LXY⁺20] used the aforementioned cross-domain approach, [LCFW20] used a meta-learning set-up and [YK20b] used a Nearest Neighbour approach for the recognition of Named Entities in text.

2.2 Language Modelling / Transfer Learning

In this thesis we are going to use two different language models for converting our textual inputs to word embedding vectors. In specific we leverage the two common Transformer implementations called "Bert-Base-Cased" and "RoBERTa" from [WDS⁺20] in our experiments.

The original CrossNER architecture used the "BERT" model already. This model was the result of [DCLT18, VSP⁺17], which uses attention to incorporate context into the embedding of each single token. These Transformer models are trained on extensive book corpora and Wikipedia texts and therefore can incorporate context better than previous approaches such as the popular Word2Vec model [MCCD13]. This enables the model to differentiate two words based on their context e.g. the bank you sit on or the bank where you deposit money.

Since, the publication of [DCLT18, VSP⁺17] Transformers have established as the most dominant pre-trained language model, according to [WDS⁺20] with over 30.000 downloaded models per day in April 2020. They are used for a variety of natural language understanding and generation tasks. This architecture surpassed the performance of existing approaches using Convolutional or Recurrent Neural Networks and the original paper [DCLT18] already established a new state-of-the-art in eleven NLP tasks including Question Answering and Named Entity Recognition.

Their word representations are the results of a model which is learned on different tasks on huge text corpora. While learning the language model it tries to predict arbitrarily hidden tokens in a sentence based on the context. In case of BERT it also trains to learn to predict if two sentences could be adjacent in a text, called next sentence prediction. The language models we used in our experiments will be fine-tuned on our target domain by using the aforementioned training targets. We refer to this training as domain adaptation or domain adaptive pre-training of the language model.

An improved learning target is to predict whole spans from the context, so the model is more dependent on the context instead of potential adjacent tokens which are part of the same entity. E.g. the initial training design can easily predict that "XXXX" is "York" in "New XXXX is also called the big apple.", since "New" is part of the entity

"New York". If the whole span "New York" is masked, the model has to rely solely on the context. [LOG⁺19] This alternative training target is leveraged in the model called "RoBERTa-Base".

2.3 Named Entity Recognition in Noisy User-generated Texts

The yearly Workshop on Noisy User-generated Text (W-NUT)[DNEL17] acts as incubator for similar research on noisy user-generated texts. The closest relative [NFM19] (domain-wise) to our research idea stems from this challenge. This challenge deals with texts from regular web pages and includes a named entity category called "product". This "product" category would by its definition include guns and drugs in our setting.

NER models in these noisy text domains are not able to reach a similar performance compared to systems focusing on standard corpora for common entities. The F1-Scores (for the definition of F-Score see section 4.2), even for common named entities categories such as Person or Location, reach less than 50%. This can be seen as quite low, compared to newspaper or Wikipedia text corpora where NER performance reaches sometimes more than 90% F1-Score.[ABV19]

The winning paper of W-NUT 2017 uses a model which includes character- and word-level features in combination with a dictionary for each entity type. Its architecture uses a CRF classifier for the final categorization, but during training it uses a two-folded learning task with named entity segmentation and categorization for learning a CNN and a Bidirectional Long Short Term Memory (BiLSTM) model for character- and word level embeddings. [AMLMS17]

This architecture was adapted by Al-Nabki [NFM19, NFAFR20] for NER in the Darknet domain. They added manually labelled Darknet market samples to the W-NUT 2017 dataset and substituted the gazetteer/lexicon with a Local Distance Neighbor (LDN) feature. The LDN feature can be described as a Nearest Neighbor algorithm, which tries to match the word embedding of each token with its closest neighbour embedding, for which a label is already known from the training instances. This model showed a better performance than Aguilar [AMLMS17] and the approach by Akbik [ABV19] which is part of the well-known Flair Framework.[ABB⁺19]

The approach of Akbik is using a special type of embedding where character level embeddings are contextualized and stored for future reference. The main idea behind this is to tackle the problem of very rare words in text corpora. The hypothesis is that rare words are usually introduced in an earlier sentence, so the word is considered to be known by user. Therefore, they use dynamic character level embeddings where each occurrence of a word will influence the future embeddings, which they call "evolving word representations". [ABV19, ABB⁺19]

The aforementioned approaches are based upon full training sets, even though the training set size of Al-Nabkis NuTOT dataset [NFM19, NFAFR20] is close to a Few-Shot

setting. However, the small amount of available training data in practice creates the need for models which can work in a few-shot scenario. Few-Shot approaches which address the challenge of transferring a well trained NER model and only adapt it to a new domain are emerging. CrossNER [LXY⁺20] and StructShot [YK20a] are examples of such architectures. Their models are trained on common NER corpora and try to transfer the abilities learned via transfer learning into other domains with different entity types. CrossNER focuses mainly on adapting the language model via domain/task adaptation (see section 2.1) and Structshot focuses on a novel nearest neighbor approach for classifying named entities.

2.4 Crowd-Sourcing

Many recent studies relied upon crowd-sourcing for the creation of Named Entity Recognition datasets within the budget and time constraints of research projects. Crowd-sourcing usually refers to a collaborative labour approach where tasks are distributed to multiple users over the internet. According to [SBDS14] the main categories of crowd-sourcing are:

- Mechanised labour: Where workers are rewarded usually on a pay per task scheme.
- Games with a purpose: Where the task is presented as a game e.g. Penguin watch ¹.
- Altruistic work: Which relies upon the goodwill of users e.g. the gun violence project ².

In this project we are going to leverage the mechanised labour platforms Appen³ and Amazon Mechanical Turk⁴ for our purpose. The scope of our project only allowed for a small-scale effort, which made us choose mechanised labour.

Crowd-sourcing can drastically reduce the manual annotation effort on side of the requester/ordering party and can increase the trust in the annotation due to the smaller probability of annotator bias, compared to having a single expert annotator. A big group of annotators with various background, which most probably only share common knowledge, will provide the annotations based on our definition which tokens shall be labelled as drug. Therefore, we assume that their annotations will be reliable if the Inter Annotator Agreement is high, as described in [ASM14]. This is important since, there are a lot of corner cases present in our texts. For example it might not be intuitive if misspelled drug names, pill imprints, co-references like "this pill" or "powder", or chemical descriptions shall be labelled as drug.

Our literature review revealed that already one of the datasets we use in this work applied crowd-sourcing for Named Entity Tagging tasks. The dataset in reference is the

¹<https://www.zooniverse.org/projects/penguintom79/penguin-watch> - last accessed 05.05.2021

²<http://gun-violence.org/> - last accessed 05.05.2021

³<https://appen.com/> - last accessed 05.05.2021

⁴<https://www.mturk.com/> - last accessed 05.05.2021

Broad Twitter Corpus [DBR16]. This is not surprising because according to the overview provided by [SBDS14] crowd-sourcing is employed for the creation of a wide range of linguistic resources.

We build our annotation process upon the guidelines defined by [SBDS14] and [FEN09]. These works provide an overview of the whole crowd-sourcing process. Starting from how to set-up the annotation guidelines, information on pricing from different previous projects up to final adjudication/data aggregation and quality assurance settings. Further information about how to set-up your task in a cost efficient manner can be found in [FSLR⁺18]. Moreover, information about effort estimation for Named Entity Tagging can be found in [GCRF20]. Effort estimation is required to ensure an ethical payment throughout a project.

Since, we figured that the annotation guidelines are the most crucial aspect of a crowd-sourcing project we conducted a thorough exploration of other named entity tagging projects such as [FMK⁺10, NBB⁺06, BBR14], and incorporated their experiences in our guidelines. A description of our data annotation process can be found in section 3.2 and details about the annotation guidelines and a detailed account of the technical aspects is provided in the Appendix in section A.2.

Approach

In this section we elaborate on the raw data used, how we structured our data annotation project via crowd-sourcing and provide statistics about the final dataset. Subsequently we present the architecture of our NER model and explain the task adaptation set-up to evaluate cross-domain transfer learning.

3.1 Datasets

Due to the highly specialized domain, we couldn't find appropriate annotated datasets for training our supervised NER models. Only a small dataset from [NFAFR20] with 851 samples from Darknet markets was available for our domain, where guns and drugs were labelled as products. We found that this dataset was too small to provide a sufficient estimation of the final model performance. It should be noted that the 851 samples still had to be separated in train, evaluation and test set. Moreover, the generic "product" entity was not specific enough for our purpose, since we wanted to be able to provide insight on a certain category of goods (e.g. guns **or** drugs). Finally, we found that [NFAFR20] mentioned that the dataset only contained entities of texts which included certain drug or gun keywords and we wanted to use an unbiased sample of drug listings.

We decided to create our own dataset to boost further research in the detection of named entities in noisy user-generated texts such as Darknet Markets. Unfortunately, annotating named entities by a team of experts was not feasible in the scope of our project, since it would exceed our resources in terms of time and budget. Therefore, we leveraged crowd-sourcing, in combination with our supervision, to label drugs in a corpus of item listings in Darknet Markets.

3.1.1 Data Sources

The Darknet data is loaded from two primary sources, the Darknet Market Archives [BCDH⁺15] and AZSecure-data [DZE⁺18].

The Darknet Market Archives contain multiple datasets about Darknet Market platforms and forums. We only used the "grams" dataset. This dataset contains nearly daily scrapes of multiple market platforms (e.g. "Agora"). We chose to use the last date where these markets were scraped "2015-07-12" and only a subset of these markets, namely: "Abraxas", "Agora", "Alpha", "ME" and "Oxygen". This dataset was only used for adjusting our language models to the target domain, called domain adaptation (see section 2.1).


For the dataset creation we used a dataset from AZSecure-data, which was scraped from a platform called "Dreammarket". At this time it was the largest Darknet market platform according to [DZE⁺18]. The data was collected from 2013 to 2017 and contained 91.463 listings of which 61.420 were found in a category associated with drugs. The dataset contains a variety of product and vendor information. In scope of this work, we were only interested in the product name and description. The item description was used for the annotation of named entities and the product name, was used to provide context to the annotators. However, other types of information were used during the pre-processing for pseudonymization purposes. The pseudonymization included removing all vendor names from the item listings, removing email addresses and telephone numbers and all links found in the dataset (those might also identify a vendors profile). A recent example for a drug item listing, which was online at the time of our project, can be seen in figure 3.1.

Our experiment design required further datasets as representatives for standard NER corpora and text corpora with noisy user-generated data. Our standard NER text corpus is the well-known CoNLL2003 NER dataset [TKSDM03], which is based on newswire texts annotated with Person, Location, Organization and Miscellaneous entities. As representatives for the noisy user-generated text datasets we chose the Broad Twitter Corpus [DBR16] and the WNUT 2017 dataset [DNEL17]. The Broad Twitter Corpus contains 9.551 Tweets with annotations for entities of type Person, Location and Organization. The WNUT 2017 dataset contains 2.295 text from various sources ((Reddit, Twitter, YouTube, and StackExchange comments) with annotations for Person, Location, Corporation, Product, Creative-Work and Group as named entity types. Furthermore, we used the extension from Al-Nabki [NFAFR20] of the WNUT 2017 dataset called "NuToT". This dataset version is extended by Darknet market listings, which advertise illicit goods.

3.2 Crowd-sourcing

Following crowd-sourcing best practice guidelines from [SBDS14] we divided the data annotation project into four stages which can be seen in figure 3.2. It should be noted that we worked with the Appen platform and later on switched to Mechanical Turk (see section 2.4), due to licensing issues with Appen.

224g of Mac 1, Supplementary Light Greenhouse, Near Indoor Quality!



Mac 1

Category: Drugs -> Cannabis - Buds and Flowers

Price (Fiat): USD 990 (€832.59 £721.35 AUD1304.18 CAD1240.21)

Price (XMR): 2.995914661824

Measurement unit: Pound

Shipping: from: United States to: United States

Views: 5

Shipping methods:

- FREE : USD 0 (XMR 0.000000000000)

Available: In stock

Vendor: 98.80 % positive / 250 reviews Disputes: 0 won / 0 lost [400 - 410 sales]

Finalize early (FE): Listing is Escrow

Vendor last seen: Today

Imported Feedback:

Empire: 99.52% / 2314 sales.

Minimum order amount: XMR 2.995914661824 (2.995914661824 for products + 0.000000000000 for shipping).

Vendor's PGP key fingerprint: [REDACTED] Show Key

Listing Description

Half Pound

224 grams

Supplementary Light Greenhouse, Near Indoor Quality!

Mac 1

Also known as Miracle Alien Cookies, this strain has had more hype in the boutique cannabis world than almost anything aside from runtz. The indica dominated high packs a serious punch and isn't for the faint of heart. This limited batch was grown with supplementary lights. It's basically indoor that's grown in a greenhouse. The unusual look and unique nose is well-known to people who have enjoyed the Mac 1.

Figure 3.1: An example for a drug item listing on a Darknet market platform called White House Market. Accessed on the 12.04.2021.

3.2.1 Project Definition

This step required us to define the NLP Problem, crowd-sourcing genre and crowd-sourcing task. Our final model was supposed to execute Named Entity Recognition and therefore the crowd-sourcing NLP problem was Named Entity Annotation. The crowd-sourcing genre was mechanised labour (see section 2.4), which is usually deployed on platforms such as Appen (earlier Crowdfunder/Figure Eight) or Amazon Mechanical Turk.

The NLP Problem of Named Entity Annotation was solved via batches of sequence marking tasks. Those tasks were completed by crowd-workers with minimal training and compact annotation guidelines. The entity annotation guidelines can be found in the Appendix (see section A.2).

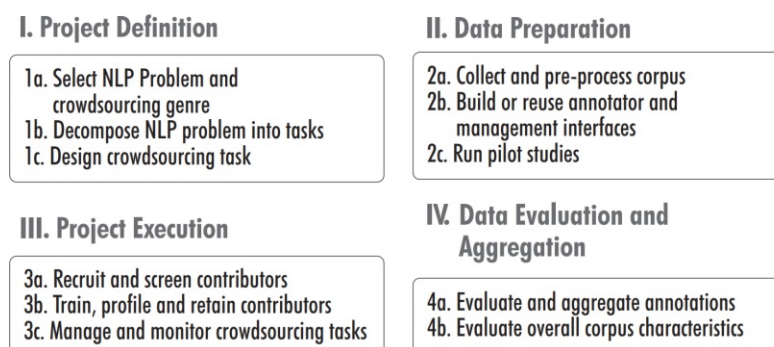


Figure 3.2: Crowd-sourcing project structure re-used from [SBDS14, p. 2]

Annotation Guidelines

The uttermost important step in designing a Named Entity Annotation task for crowd-sourcing was the definition of what should be labelled. [FEN09] Without precise guidelines on which tokens shall be labelled, we wouldn't be able to consistently reproduce our annotation. Because of that, we specified a set of rules in order to consistently define which tokens were supposed to be labelled as a drug. These rules can be found in the Appendix in the section A.2.1 "Annotation Guidelines for Experts". These axioms were refined to a more simplistic form, so crowd-workers would be able to understand them in the short amount of time, which is used to prepare them for crowd-sourcing tasks. This simplistic form is also presented in the Appendix (see section A.2.2).

The task description contained a more detailed version in the beginning to explain the task and a small summary / set of heuristics next to the annotation window. This was later changed to only a small heuristic with access to the exhaustive description upon request, when changing to MTurk. The heuristic version is needed to mitigate the risk that annotators won't read our expressive set of rules due to its length and work on the task without having read any description at all. The detailed version was always available throughout the annotation process by just expanding the overview.

The annotators were presented with the name of the item listing as context information and with the item description as annotation target. An example can be seen in figure 3.3.

The only entity type labelled was the type "Drug". Annotators were supposed to find occurrences of legal, illicit or prescription drugs in the descriptions of items sold on Darknet platforms. The Food and Drug Administration (FDA) defines a drug as.: [FA21]

1. "articles intended for use in the diagnosis, cure, mitigation, treatment, or prevention of disease"

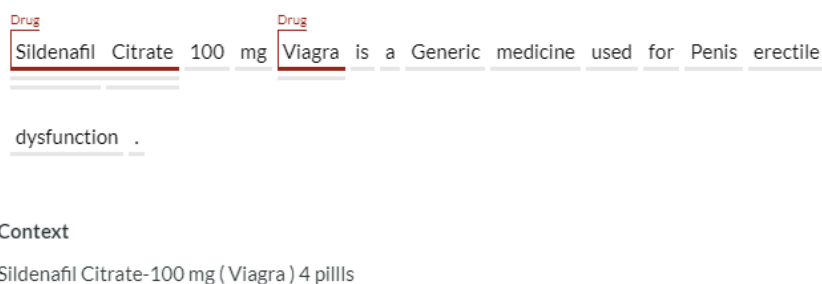


Figure 3.3: Screenshot from the Appen labelling tool.

2. “articles (other than food) intended to affect the structure or any function of the body of man or other animals.”

During the creation of the annotation guidelines, we used this definition of the FDA as basis for decision making. Moreover, the entities labelled were required to uniquely identify a drug according to the FDA definition. Each annotation should clearly identify a drug on its own. In the example "Charley’s mellow Sleeper Bars infused with THC" we would label only the token "THC" as a drug, since it clearly identifies an active agent of Marijuana. "Charley’s mellow Sleeper Bars" could be a drug item as well, however it does not clearly identify any specific drug entity. Therefore, we decided to solely label entities which fulfill this constraint in order to mitigate the annotation of tokens we consider irrelevant in scope of drug recognition for law enforcement agencies.

3.2.2 Data Preparation

The DreamMarket data (see section 3.1.1) can be accessed in form of a relational database (MySQL). We exported all item listings associated with a drug category. Of the aforementioned 61.420 drug listings, we extracted 45.446 items via the MySQL Workbench Frontend. Only 11.674 of those items remained after we removed item listings:

- which contained exactly the same textual description as another item listing (duplicate removal - 45.446 down to 20.434)
- with a description categorized as non-english text by Google language detection [Shu10] (20.434 down to 18.332)
- with very long (>3000 character) or short (<30 character) text as description (18.332 down to 16.844)
- where the description contained exactly the same non-numeric characters in the first 100 characters as another item listings description, since that indicates that probably only the amount of the drug changed in this listing compared to other listings. (16.844 down to 11.674)

Subsequently, vendor information such as telephone numbers, email addresses or vendor names have been removed and replaced by random values to be compliant with EU-GDPR

guidelines. We used the python port of Google’s phonenumbers library for detecting phonenumbers ¹, regular expressions for email addresses and the vendor names were provided in a structured format by the DreamMarket [DZE⁺18] SQL dump. Further pre-processing was conducted with the NLP framework Stanza [QZZ⁺20]. The Stanza pre-processing included the application of Stanza’s tokenization module and the removal of special characters and links. We removed the special characters and links via a custom module using the function in Listing 3.1.

Listing 3.1: Pre-Processing function used in Stanza module, for the removal of links and special characters.

```
def remove_unwanted_elements(text):
    final_text=text
    final_text = re.sub(r'https?:\\/\S*[\r\n]*', '', final_text)
    final_text = re.sub(r'\S*.onion\S*[\r\n]*', '', final_text)
    final_text = re.sub(r'[+!~@#%$%^&*()=}{\[\];;<.>?\`"]', '', final_text)

    #Only for MTurk replace ', ' with its unicode descriptor and
    # re-convert it after the annotation process.
    final_text = re.sub(', ', '&#44', final_text)

    final_text = re.sub(r'[-]+' , '-' , final_text)
    final_text = re.sub(r'[_]+' , '_' , final_text)
    return final_text
```

For the creation of annotator interfaces we used standard sequence marking templates from Appen and Amazon Mechanical Turk (MTurk). These showed a sufficient quality and ease of use to be feasible for our project. Pilot studies for annotating examples with early versions of the annotation guidelines were conducted with non-native English speakers as crowd-workers (see section 3.2.3). These pilot studies highlighted the importance of simplistic annotation guidelines in terms of vocabulary used as well as simplicity of the instructions. Furthermore, we found that screenshots of the annotation tools with labels eased the understanding of our task for annotators, compared to textual examples.

3.2.3 Pilot Studies

Led by principles defined in the Appendix in section A.2, we conducted initial experiments to refine the annotation guidelines. Prior to evaluating our annotation guidelines with possible test annotators we manually labelled 500 item listings by our own. We did this to find possible inconsistencies in our annotations guidelines and corner cases which are not well defined when following our annotation guidelines. Based on the problems discovered during this initial annotation we refined our rules to ensure a consistent annotation and a sufficient coverage of possible corner cases.

We set up a prototype on the two common platforms Mechanical Turk and Appen used in other literature when conducting crowd-sourcing. [JMGB20, FLRS⁺15, BDR17,

¹<https://pypi.org/project/phonenumbers/> - last accessed 17.05.2021

FSLR⁺18] After an initial comparison of the platforms we decided to use Appen due to its performance evaluation features. This enabled us to use gold standard annotations to evaluate the crowd-workers performance. Mechanical Turk did not offer this feature, so one would have had to implement it on his/her own or rely on inter-annotator agreement as performance measure. We wanted to mitigate the risk of having a high agreement by people who did not read our annotation guidelines and only tried to solve the task intuitively. They may only annotate the most intuitive expressions seen as "drug" such as "cocaine" or "Marijuana" as drugs and ignore all ambiguous terms, which might or might not refer to a drug. We assumed this could have caused massive efforts to exclude ill performing crowd-workers, compared to Appen where this bench-marking of annotations against gold standard data is a standard feature.

Unfortunately, we had to switch from Appen to Amazon Mechanical Turk due to licensing issues. Appen required us to pay additional 5000\$ for a license after the first 1000 annotations, which was way above our budget.

3.2.4 Project Execution

We executed two data annotation projects, since we switched from Appen to Mechanical Turk. In the end we gathered over 4.000 annotations from Appen workers and over 7.400 annotations from Amazon Mechanical Turk workers. We continuously monitored the annotation quality based on the Inter Annotator Agreement (IAA - see section A.1.1) and conducted manual reviews of annotation samples from each single worker. Furthermore, we bench-marked the annotation quality at later stages of the project based on the IAA with trusted annotators. After each run we employed quality improvement measures, if necessary. We focused on refinements of data annotation guidelines, but we also evaluated measures such as changing the reward per task, max. task assignment time or number of annotations. A detailed description of the measures employed due to our experiences in the project execution can be found in the Appendix (see section A.1).

3.2.5 Data Evaluation And Aggregation

We explored different aggregation / adjudication techniques used in the literature. According to the comparative study of [SBDS14], the majority of crowd-sourcing papers used majority voting. A variation of this technique is the default method of Appen. Appen extends the majority voting principle by weighting each annotators vote with the "trust" score of the annotator. This "trust" score is based on his/her accuracy on test questions. The method is called "tagg"².

Another possible option was to improve recall by considering all annotations and conducting a final review by experts used for the BTC dataset ([DBR16] - see section 3.1.1). However, we found this to be inefficient, since we are not solely focused on recall.

²<https://success.appen.com/hc/en-us/articles/360050002751-Guide-to-Text-Annotation-Aggregation> - last accessed 05.05.2021

In case of the data annotation with Appen, we used a variant of Appens default strategy combined with a final expert review to further improve recall and precision. In case of MTurk, we used majority voting with a subsequent expert review. For the final expert review, we converted the exports from Appen/MTurk to the format of labelstudio [TMS⁺21] and conducted a manual review of the aggregated annotations.

The final Inter-Annotator Agreement (IAA) of both parts of the annotation can be found in Table 3.1. We found that the annotations provided by Amazon MTurk were more consistent than the ones was provided by Appen crowd-workers. Following [VG05] and [HR05] we used the Mikro F1-Agreement as primary IAA measure in addition to Cohen’s Kappa as IAA measure, which is used for comparative reasons. We prefer F1-Agreement over Cohen’s Kappa, since [DLL⁺12, HR05] state that Cohen’s Kappa requires the number of negative cases, which is not known in a NER scenario. If a sufficient amount of negative cases is present, they showed that Chohen’s kappa is close to the value of the F-Measure. Moreover they argue that NER datasets are highly imbalanced with a much larger amount of negative cases than positive cases, which is another problem with Cohen’s Kappa. Compared to a crowd-working performance study for noisy user-generated texts [JMGB20] we even outperformed most of their annotation experiments (compare table 3.1 and A.2) in case of the Amazon MTurk annotation quality.

IAA	Mturk	Appen
Cohen’s Kappa	0.76	0.43
Mikro F1 Agreement	0.79	0.55
Makro F1 Agreement	0.78	0.60

Table 3.1: Inter-Annotator Agreement measures for the overall annotations by crowd-workers of Amazon MTurk and Appen.

Final Review statistic	MTurk	Appen
% of characters added by final reviewer	8.56	25.81
% of characters deleted by final reviewer	3.51	0.89
% of spans added by final reviewer	13.19	27.49
% of spans deleted/alterd by final reviewer	8.65	10.45

Table 3.2: Performance evaluation of the review process

In table 3.2 we can see that we reduced the annotation effort significantly, in comparison to manually annotating all item listings on our own. Annotations from users of the Mechanical Turk platform were more compliant to our definition of the concept "drug" than annotations from users of Appen. We assume that this could be due to the different targets of the crowd workers. Appen crowd workers only work against the test questions. We assume that these workers tend to solely mark clear examples of drugs and are not willing to take the risk of marking any corner-cases. Therefore, their annotations

Total documents	Annotated spans	Unique spans (ignoring casing)	Unique spans (including casing)
3,507	14,934	3,048	3,739

Table 3.3: Key facts of datasets

were quite "conservative" and over 99% overlapped in terms of characters (see row 2 in table 3.2). Crowd-workers on Amazon tended to annotate corner-cases more often. This resulted in more rejection of their spans, but a substantially higher percentage of the final annotations were done by the AMT workers, than by Appen Workers. This is in line with our main reason for using crowd-sourcing, that the requester has to do less manual annotations on his/her own and that the final annotation is regarded as the common view on the subject matter based on the requester's definition.

3.3 Final Dataset Statistics

The final dataset, called DreamDrug, for Drug NER in Darknet texts contains 3507 item listings with 364.003 tokens. The dataset was split into:

- Training dataset: 2244 data points
- Evaluation dataset: 561 data points
- Test dataset: 702 data points

The training dataset was used for optimizing the weights of our models during the training phase. Subsequently, the performance of our different models was assessed and compared based on the evaluation set. Finally, the performance of the best models from the evaluation set was estimated by predicting the examples in the test dataset.

In table 3.3 and 3.4 we present descriptive statistics about our dataset DreamDrug.

	Min.	Max.	Mean Value	Median Value
Words per document	3	534	103.79	61
characters per document	25	2,930	572.60	334
words per span (spaces per span)	1	20	1.88	2
characters per span	1	80	12.06	10

Table 3.4: Descriptive Statistics of dataset

About 20% of the examples are considered long texts with a length of 1000 to 3000 characters. The other 80% are shorter than 1000 characters. In terms of tokens the 80% quantile would be at 176 tokens (mean character length of a 176 token text is 1006.11). An overview of token and character length per text, can be found in figure 3.4. An overview of the most frequent spans labelled as "drug" can be found in the word-cloud in figure 3.5.

3. APPROACH

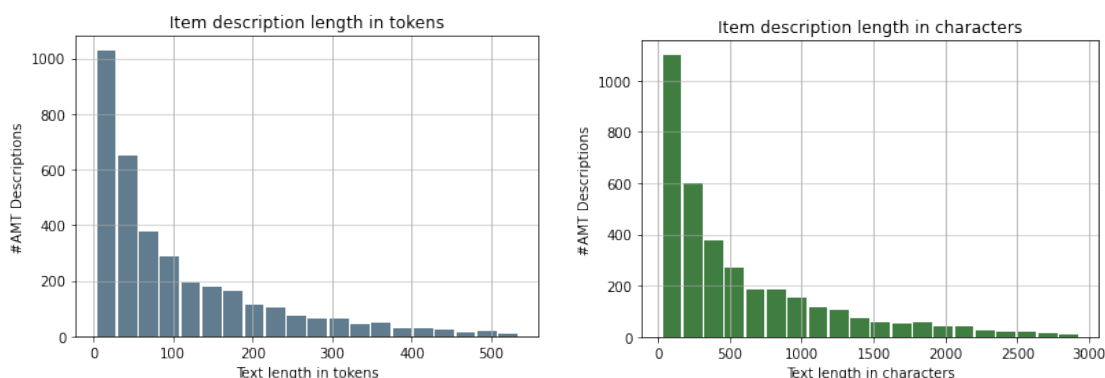


Figure 3.4: Lengths of drug item listings in final dataset

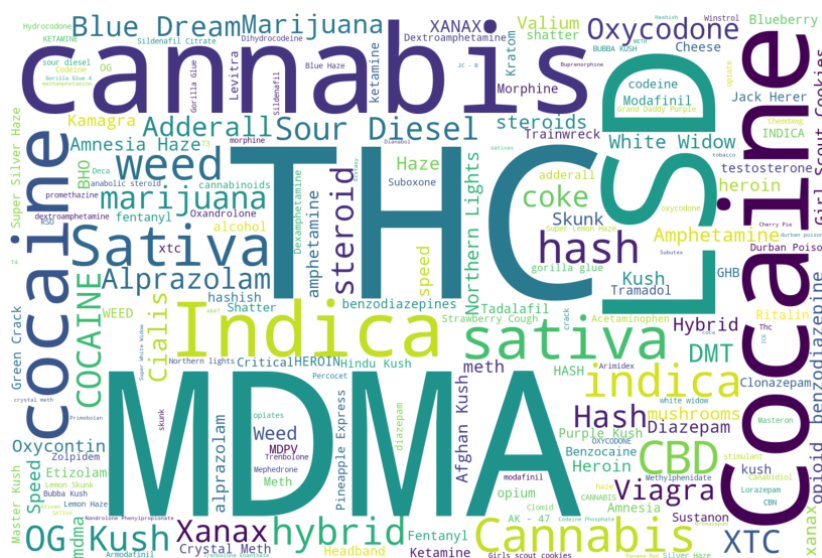


Figure 3.5: Word-cloud of the most occurring spans annotated as drugs, scaled by their occurrence.

While looking at the word cloud in figure 3.5 one might get the impression that most drugs are quite common and the expressions used are well known. However, Figure 3.6 shows that in fact only about 54% of the drugs appear more than 20 times and the remaining terms are rather rare. When evaluating a small sample of 50 examples, we found that they were caused by specific rare terms (e.g. Strain names for Cannabis), spelling variations of common drugs (e.g. "morphine hydrochloride"), misspellings (e.g. "oxymorphone") or specific chemical descriptors (e.g. "n-acetyl-p-aminophenol"). It should be noted that spelling variations and specific rare terms were the most common reasons.

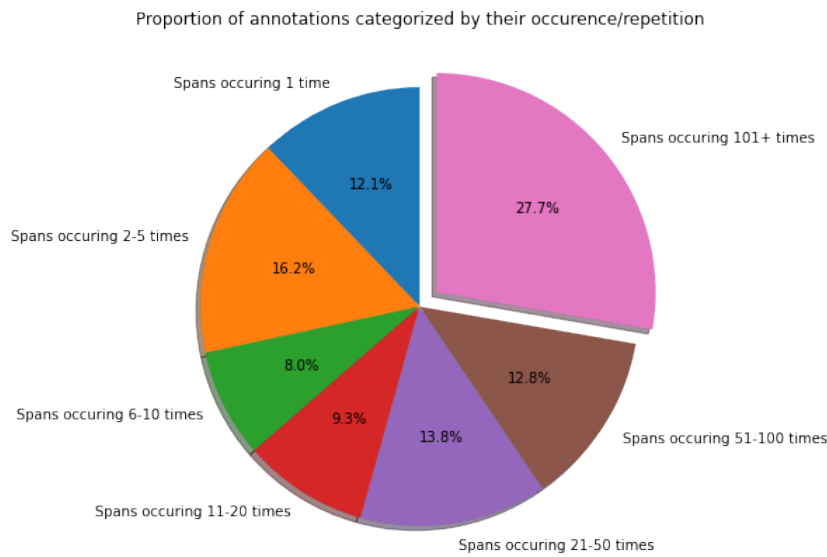


Figure 3.6: Relative contribution of spans, clustered by the amount of their occurrence, to the total amount of annotations.

3.3.1 Limitations

To this point in time there are still a few options for improvement in regard to our dataset DreamDrug. We annotated our data according to the drug definition of the FDA with respect to unique drug identifiers (see section 3.2.1). Nonetheless, there are open issues due to the ambiguity in natural language and especially in user generated inputs. We found that the following examples are still open for discussion:

- Pollen / Polm for hasish - since it refers to pollen of plants, which on its own does not act as clear identifier of a drug. But in the context of Darknet market items it often does.
- Fishscale / Colombian Fishscale / Colombian Flake - Those those words are often used as descriptor or adjective of Cocaine. However, some times it was used as standalone entity, and therefore could be seen as a reference to cocaine.
- Ecstasy Names like Green Dom Perignon / Sim Cards / Red Bulls - those names on their own do not clearly identify a drug (e.g. Yellow schoolbuses or Green Hulks are Xanax and not XTC). Therefore, we did not label them as "drug". One might argue, that in this context they usually identify Ecstasy due to the common pattern of colour and a Brand Logo.
- Testosterone - we labelled artificially created steroids like Deca, Nandralone or Testosterone Enthanate. On the contrary we did not label naturally occurring elements of the human body like dopamine. Testosterone alone was often a quite difficult corner case since it is often a naturally occurring hormone, but also sometimes a synthesized drug.

- Blotters - even though it is the form of a drug like bud, pill or powder, one might argue that blotter is such a unique word that it clearly identifies LSD. However, we did not label it, since it is the form of a drug.
- Mushroom - Even though we were initially quite sure that "mushroom" in the context of Darknet markets clearly identifies the drug "magic mushrooms" or "Psilocybin", we found that it can also refer to the form of drug. For consistency reasons, we continued to label mushrooms as drug.

Another difficulty is to find a common agreement of span boundaries. "Viagra Sildenafil" can be seen as a single drug or as the brand name of the drug in the first term and the active agent of the drug in the second term. During our supervision / review of the data we had in mind to separate drug concepts from each other. Therefore, in our opinion the correct case is to separate them. In practice, this is quite difficult. "Haze weed" for example can refer to the specific type of cannabis called "haze" and the "street" slang of cannabis "weed". Representing two concepts means it could be separated, but one might disagree and say that "haze weed" is a single entity since the entity "haze" describes the entity "weed" in more detail and is therefore part of it.

3.4 Other Datasets

3.4.1 Distantly Supervised Drug Dataset

In order to automatically annotate drugs in this additional dataset, we leveraged a technique called distant supervision. This refers to annotating raw data by looking up terms of the text corpus in an existing Knowledge Graph. Subsequently, properties of the matched concepts in the Knowledge Graph can be explored with a query language called SPARQL [hom11]. A detailed description of Knowledge Graphs and Distant Supervision can be found in [SACM19].

As basis for our distantly supervised drug dataset we used Wikipedia's Python API³ to download about 2700 mostly drug related articles. In order to exclude irrelevant parts of the text we removed parts of the text e.g. "References" or the "See Also" section.

We iterated over all sentences from those articles to link contained entities to potential matches in the Knowledge-graph DBPedia [ABK⁺07] via the Spotlight API⁴. Subsequently the entities found were checked for relations which indicate that an entity is a drug via SPARQL queries. Our focus was on illicit drugs with relations such as "dbp:legalUs", which represents the legal status of drugs in the USA. It should be noted that the quality of this distantly supervised dataset is quite poor. The terms "Cocaine.", "(LSD)" or "MDMA,", for example were not considered as drug due to the dot, brackets or comma contained in each term. This was especially problematic at lists e.g. in "In a hospital environment, intravenous clonazepam, lorazepam, and diazepam are first-line choices" only diazepam is recognized as drug. Moreover, the chosen SPARQL query wasn't able

³<https://pypi.org/project/wikipedia/> - last accessed 05.05.2021

⁴<https://pypi.org/project/spacy-dbpedia-spotlight/> - last accessed 05.05.2021

to detect drug categories (e.g. "opioids"), slang names for drugs or advanced chemical descriptors. Therefore, further efforts for pre-processing measures and query refinements would have been required, in order to provide dataset with sufficient quality.

3.4.2 Pre-Training Text Corpora

Following [DCLT18] we fine-tuned our language model (see section 2.2) on our target domain. Since we are using other transfer-learning techniques as well we call this step domain adaptation (see section 2.1). Therefore, we need texts from the target domain or other textual resources which include relevant content to our target domain.

We used various options as Domain Adaptive Pre-training Text (DAPT) for fine-tuning our language models. In case of our wikipedia illicit drug corpus, we only kept sentences which contained drug entities. We hoped to improve the effectiveness of the DAPT corpus by removing irrelevant parts of the text as seen in [LXY⁺20]. The evaluation if a drug entity is present or not is derived from the results of our distant supervision process described in section 3.4.1.

In order to prepare our model for real world usage on Darknet markets we use examples of item descriptions from Darknet Markets. We included all drug-related samples from the Dreammarket which were not used for the annotation process (see Section 3.2.2). Moreover we used all item descriptions from the Grams dataset (see section 3.1.1). It should be noted that the Grams dataset contains item listings of other product categories as well, so the advertised goods can be guns, pornographic content or fraud services. All of those three text corpora were evaluated as options for DAPT, to assess if they actually improve the prediction performance. The DAPT for fine-tuning our language models called "All" is a mixture of all aforementioned texts.

3.5 Model Architecture

This study was based on the original model architecture of the CrossNER paper [LXY⁺20]. The model architecture itself was quite similar to the original proposal from [DCLT18]. We used a Transformer architecture for our word embeddings and two linear layers on top, each with dropout, on top of the Transformer for the classification of Named Entities. We adapted this model for our purposes and evaluated minor improvements for the model architecture, which are presented in this section. A visual representation of our architecture can be found in Figure 3.7. In case of the Few-Shot setting we leveraged only a single linear layer.

3.5.1 Transformers

In this thesis we use the Transformer implementations from Huggingface [WDS⁺20] in PyTorch as seen in [LXY⁺20]. As initial baseline we use the BERT-Base-Cased model and as optional improvement we decided to use RoBERTa-base, due to their span based language model training and internet text focused pre-training corpora (see section 2.2).

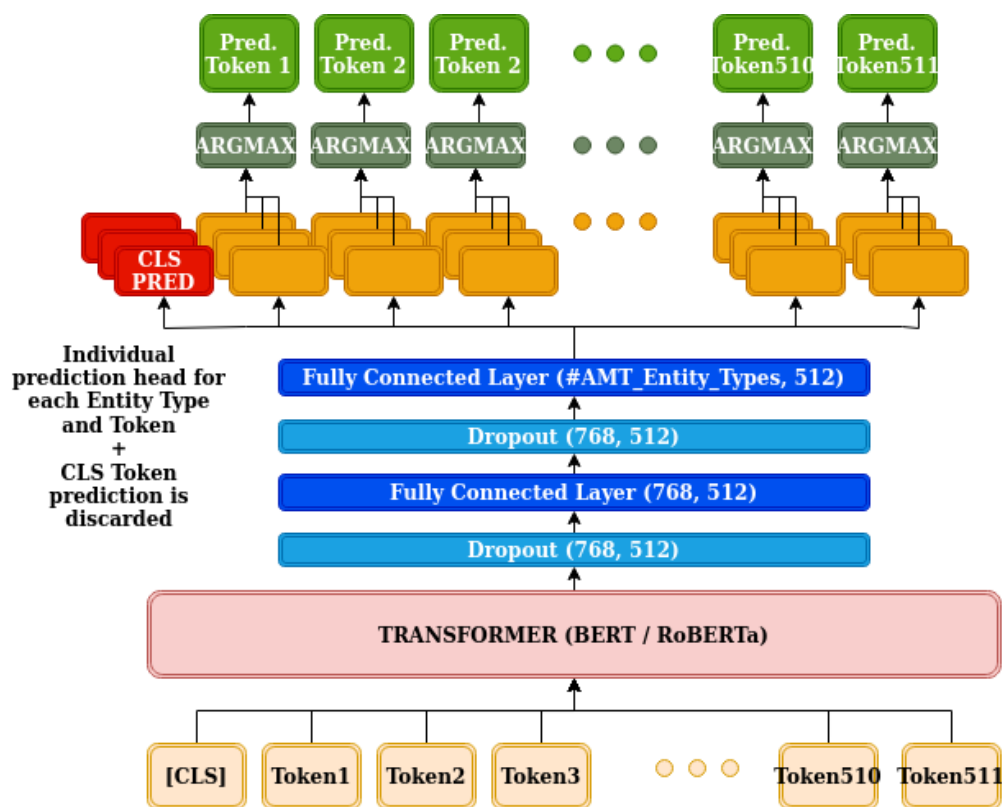


Figure 3.7: Model architecture used for our NER task

We evaluated 4 types of Domain Adaptive Pre-training Texts (DAPT) (see section 3.4.2) to adapt our language models to the target domain

- Dreammarket description texts, which were not used for annotation.
- Grams description texts, a variety of darknet item listings spanning over all domains.
- Wikipedia distantly annotated dataset
- All of them combined

3.5.2 Classification Layer

For the classification of Named Entities we leveraged two feed forward layers as classification model on top of the Transformer embeddings. This model had two prediction heads for each entity type, including the "O" - "other" entity type. Due to over-fitting of our model in initial experiments, we evaluated the options for weight-decay/L2-regularization and dropout. In initial tests we found it particularly hard to find the right hyper-parameters for L2-regularization, but could easily employ dropout before each layer with the expected regularization effect. Therefore, we decided to use a dropout layer similar to [EBNA16] to prevent over-fitting.

3.5.3 Data Pre-processing

Textual inputs for Named Entity Recognition models are often cut into sentences and predicted separately. In our setting of small item listing descriptions we found that this setting would remove relevant context. The content of these small item descriptions is strongly connected and removing adjacent sentences would disable our Transformer language model to leverage the context from other sentences. Therefore, we evaluated different pre-processing options for the text length.

Due to the model design of our language models, the maximum input length was 512 tokens. According to Figure 3.4 this should not impose a problem, since only a handful of texts have more than 500 tokens. Unfortunately, tokens which are fed into a Transformer model and text tokens (words) are not referring to the same concept. BERT and RoBERTa (see section 2.2) need token-ids as input, which are converted from plain text to such token-ids by their individual tokenizers.

These tokenizers are not equal to common tokenization modules (e.g. from STANZA [QZZ⁺20]). Common tokenizers focus on separating whole paragraphs of a text into sentences and these sentences into separate words. In opposition the tokenizers of our Transformer models are subword tokenizers⁵. The principle behind subword tokenization is that common words shall not be split into subwords to preserve their meaning, but rare terms which would not be known to the model shall be split into meaningful subwords. Moreover, the tokenizer converts words from their textual form into the aforementioned token-ids. Since, one word might result in multiple sub tokens, the final amount of tokens can be higher and will often be longer than 512 tokens (the maximum input length of our language model).

In order to tackle this issue we evaluated four options:

1. Cutting off tokens after the 500th token (about 4-5% of the tokens in train/evaluation/test set).
2. Separating long texts at the first occurring sentence boundary after the 400th token and predicting those two text fragments separately. Sentences boundaries are set according to STANZA's tokenization module [QZZ⁺20] (about 4-5% are affected).
3. Splitting all texts into sentences according to the sentence boundaries defined by our STANZA pre-processing [QZZ⁺20].
4. Removing all over-length texts from the dataset (excludes about 5% of all item listings).

In order to decide upon this matter in an empiric way, we conducted experiments with all four options. The results were based on our best model, including our best hyper-parameter setting at this point in time and are presented in table 3.5.

The best results were achieved by the method excluding tokens after a certain length. Tokens in subsequent parts of text are harder to predict, since they often lack essential

⁵https://huggingface.co/tokenizers/tokenizer_summary.html - last accessed 20.05.2021

Splitting Method	F1	Prec.	Recall
1 - cut off	81,37	84,45	78,51
2 - split into big chunks	81,33	84,96	78,00
3 - split into sentences	76,19	78.89	73,69
4 - exclude long texts	80,77	82,13	79.46

Table 3.5: Results from splitting method evaluation

context. Drug names are often introduced in the beginning of the item listing and all further splits would miss that information. E.g. in "Exceptionally flavoursome and potent, Blue Haze from Zamnesia Seeds... " we would miss that "Blue Haze" is actually the strain name and "Blue" is not just an adjective to "Haze", which shouldn't be labelled.

When using the splitting method number two, splitting texts into big chunks, the results slightly decreased. Moreover, method number three, splitting text into sentences according to Stanza [QZZ⁺20], decreased the prediction performance significantly. We assume that both of those decreases were caused by the lack of context.

Due to the small percentage of affected tokens (about 4-5%) we decided to go with the cut-off method. In case one might be required to handle all lengths of text, we would recommend to either include the product name as context information for each split or to split with overlaps. This should mitigate performance decrease even with longer text, since the context can usually be inferred from the product name as well. This advice is stated for practitioners, which want to use these models for drug detection on raw data scraped from Darknet markets.

3.6 Name Entity Recognition

3.6.1 Task adaptation NER

Following the approach from [LXY⁺20] we trained our model previously on a different NER dataset, where it is supposed to learn the NER task. Subsequently, we trained our model on our final target dataset. We refer to the domain of a dataset used for the initial pre-training as source domain and to the domain of the final target dataset as target domain. We call this technique task adaptation as described in section 2.1. We aim to answer our research questions about if and how the difference in text structure (noisy user generated text / well formatted grammatically correct text) will impact our final model performance, by those experiments.

Our hypothesis is that learning the NER task on a noisy (similarly noisy) source domain will improve the models performance, due to text structure similarity.

To evaluate the impact of different domains we are going to explore the following settings:

- Only train on target domain. - I.e. directly train on the DreamDrug dataset.

- Train on well structured (unrelated) source domain and then on noisy target domain.
- I.e. train initially on the ConLL2003 dataset [TKSDM03] and subsequently on the DreamDrug dataset.
- Train on well structured (related) source domain and then on noisy target domain.
- I.e. train initially on the distantly supervised Wikipedia drug corpus (see section 3.4.1) and subsequently on the DreamDrug dataset.
- Train on a noisy (unrelated) source domain and then on a noisy target domain.
- I.e. train initially on the broad twitter corpus [DBR16] or WNUT [DNEL17] dataset and subsequently on the DreamDrug dataset.
- Train on noisy (related) source domain and then on noisy target domain - I.e. train initially on the the WNUT dataset enhanced by drugs and guns as category "product" and subsequently on the DreamDrug dataset.

Experimental Evaluation

4.1 Experiment Design

We conducted two different stages of experiments. In stage one we evaluated options for the choice of pre-training corpora (DAPT, see section 3.4.2), language model and the dropout hyper-parameter. The result of stage one was the parametrization for a well performing model, using common performance improvement measures. Stage two was about the evaluation of task adaptation measures (i.e. pre-training on a different NER task) with various source domains. Subsequently, we were able to answer our research questions based on the results we achieve in this section.

Furthermore, we shined light onto the prediction capabilities of these models in case we could only leverage a few examples as training dataset. For this reason we selected 100 examples from the training set and initially trained our models on those. Based on this small sample we were able to estimate how successful our model might be in a so called few-shot setting. This setting is usually closer to practical scenarios due to the lack of proper datasets in specialized domains.

In order to compete against a common and easy to implement model we were benchmarking our results against FLAIR [ABV19]. This NER model used Transformer-based word embeddings, without any language model fine-tuning on the target domain. In addition a sub-token/character based embedding was leveraged, which was continuously updated during training and prediction. The architecture of FLAIR is highly sophisticated and stems from the current generation of NER models. However, it lacks the pre-training/fine-tuning and transfer learning measures. This lack of performance improvement measures was intended to measure how well our model competes against an off-the-shelf solution.

For the evaluation of our stage one experiments we executed a full experiment grid with the cross-product of all possible hyper-parameters. We trained all our models for 10

epochs in the Few-Shot scenario and 5 epochs when using the full dataset. The only exception were FLAIR models, which were trained for 50 epochs. Our final experiment design has 220 (2x2x5x11) hyper-parameter settings in order to cover every possible combination:

- Training dataset size:
 - 100 (Few-Shot scenario)
 - 2244 (Full training dataset)
- Language Model:
 - RoBERTa-Base
 - BERT-Base-Cased
- Domain Adaptive Pre-Training Corpus (DAPT):
 - None
 - Dreammarket
 - Grams
 - Wikipedia
 - All Combined
- Dropout: 0 - 0.5 in steps of 0.05 (11 values)

The results presented in section 4.3 and 4.4 were all evaluated on the validation set (except for the final test set results in table 4.5 and 4.6) and not on the final test. We were still comparing the performance of models for optimization reasons and therefore were not allowed to use the final test set for performance evaluation. Information about the dataset splits and general dataset statistics can be found in section 3.3.

4.2 Evaluation Metrics

The final prediction results shall indicate named entities using the BIO notation. This means the sentence “Selling the best White Widow on the market!” will have “White” annotated as “B-Drug”, a tag which indicates the beginning of a named entity. “Widow” will be annotated with “I-Drug”, a tag which indicates a token inside a named entity. All other tokens which do not represent a drug are labelled with "O" for "Other" entity.

In this study we evaluate our models with three metrics precision, recall and F1-Score:

$$Precision = \frac{TP}{TP+FP} \quad Recall = \frac{TP}{TP+FN} \quad F1 = \frac{2*Precision*Recall}{Precision+Recall} = \frac{2*TP}{2*TP+FP+FN}$$

Where:

- TP - Are true positives. This means the token’s gold annotation was equal to its prediction and unequal to "O" - for other entity (see section 1.2.1)
- FP - Are false positives. This means the token’s gold annotation was "O" for other entity but it was actually predicted as entity of another type.
- FN - Are false negatives. This means the token was predicted as "O" for other entity, but its gold standard annotation was actually another entity type.

In our work we re-use the metric chosen by [LXY⁺20], the CoNLL2003 NER metric. "Precision is the percentage of named entities found by the learning system that are correct. Recall is the percentage of named entities present in the corpus that are found by the system. A named entity is correct only if it is an exact match of the corresponding entity in the data file." - as stated in the original CoNLL2003 paper [TKSDM03]. Other available evaluation metrics in the NER discipline could be e.g. from the Message Understanding Conference (MUC - [GS96]), Automatic Content Extraction (ACE - [DMP⁺04, ACE08]) or the Computational Natural Language Learning (CoNLL - [TKSDM03]) conference. For further information on possible evaluation metrics we recommend the blog from David S. Batista ¹.

In cases of indexing and knowledge integration [FEN09] recommends to focus on recall. In those cases we are interested in all kind of variations of entities which can occur in user generated texts, but still refer to a specific drug entity. "For this type of application, the main point is not to highlight all the mentions of an entity in a document, but to identify which documents mentions which entity. Therefore, precision has to be favored over recall" -[FEN09].

While we agreed with the statement that our primary concern was to find out which document refers to which entity, we were not convinced that favouring precision over recall will actually benefit this target. Since the majority of our tokens, could be identified without context, we hypothesized that this measure would lead to a setting where mostly tokens, which cannot be mistaken for anything else than a drug will be labelled. Our model would degenerate in this case to a gazetteer, because only straight-forward drug descriptors would be contained in our annotation. For this reason, we stuck to the F1-Score, where precision and recall are weighted equally. Alternative F-Score parametrization for the model evaluation with a different Beta-Parameter could adapt the model better for different targets in the future.

4.3 Results - General Performance Evaluation

After conducting the experiments for stage one we analyzed their results. When looking at the differences in prediction performance stemming from the choice of language model, we found that both available options performed quite similar. The Bert-Base-Cased model (see section 2.2) achieved better results for the few-shot scenario with 100 training samples and on the full training dataset the RoBERTa-Base model (see section 2.2) achieved better results. In both scenarios the optimal dropout parameters were quite high and the optimal text corpora for domain adaptive pre-training (DAPT) was the combination of all available texts. The results of an initial parametrization without any performance improvement measures and our best models can be found in table 4.1.

It should be noted how much bigger the impact of improving the word embeddings (DAPT) is in the few-shot scenario, compared to the improvements achieved when trained

¹http://www.davidsbatista.net/blog/2018/05/09/Named_Entity_Evaluation/ - last accessed 05.05.2021

<u>Few-Shot Scenario</u>					
Model	DAPT	Dropout	F1-Score	Precision	Recall
Best Flair moel	None	0.0	59,37	NA	NA
BERT Baseline	None	0.0	66,69	68,03	65,41
BERT + DAPT	All	0.0	70,65	76,06	65,96
BERT+ Dropout	None	0.5	65,4	65,65	65,14
Best BERT model (DAPT + Drop.)	All	0.5	71,37	73,48	69,38
Best RoBERTa model	All	0.05	70,98	75,61	66,89
<u>Full Training Dataset</u>					
Model	DAPT	Dropout	F1-Score	Precision	Recall
Best Flair moel	None	0.0	72,84	NA	NA
RoBERTa Baseline	None	0.0	80,49	83,26	77,89
RoBERTa + DAPT	All	0.0	81,39	82	80,78
RoBERTa + Dropout	None	0.5	80,36	85,66	75,67
Best RoBERTa model (DAPT + Drop.)	All	0.5	82,79	83,94	81,67
Best BERT model	All	0.5	82,17	85,01	79,52

Table 4.1: Model performance during hyper-parameter tuning using 100 rows for training in the Few-Shot scenario in the upper part and the full training set in the lower part of the table. Bold font marks the best model for each setting.

on the full training set. We assume that if less data is available, it is crucial to optimize the word embeddings.

In order to provide more insight onto how well both types of language models perform, you can find the model which used the alternative language model in table 4.1 right below the best model for each scenario.

The results achieved by our model were substantially higher than the results from our competitor model, an off-the-shelf FLAIR model [ABB⁺19], which was solely trained on our training dataset. The results in table 4.1 show that our model achieved exactly 12 points more in terms of F1-Score in the Few-Shot scenario. In case of the full training dataset the FLAIR model still couldn't compete with the performance of our models. To be precise, we experienced a performance drop of about 10 points in F1-Score from our best model to the best FLAIR model.

The recall and precision values are missing for FLAIR models, since those values are only reported for the final test set and not during the training process upon the evaluation set. In modern versions of FLAIR e.g. 0.8.0 such values can be reported, however we used the older version 0.6.0 where those values were not available.

4.4 Results - Task Adaptation

In stage two we conducted experiments to evaluate potential competitive edges by pre-training our NER models on different NER corpora. The best hyper parameters found in stage one were used for all further experiments. In stage two of our experiments we were especially interested if the text structure similarity is correlated with the final performance and if we were able to achieve a performance improvement by pre-training on different NER Tasks.

Pre-Training Dataset	LM	DAPT	Drop.	F1	Precision	Recall
None	BERT	None	0.0	78.61	84.39	73.58
None	BERT	None	0.5	79.59	82.14	77.19
None	BERT	All	0.0	81.47	86.05	77.36
None	BERT	All	0.5	82.17	85.01	79.52
CoNLL	BERT	All	0.5	79.98	85.89	74,8277
BTC	BERT	All	0.5	81.76	85.47	78.36
W-NUT	BERT	All	0.5	80.48	84,35	76.95
NuToT	BERT	All	0.5	82.58	86.76	78.79
Wiki	BERT	All	0.5	78.17	79.89	76.52

Table 4.2: Full Training Set: Results after training on a different NER dataset in advance. Bold font marks the best and second best model. The second best model achieves a comparable performance, without any pre-training on a different dataset, and is therefore highlighted as well.

Our results showed that we were not able to achieve a similar performance increase as presented in the original paper [LXY+20]. In their work, they trained the model on the CoNLL2003 NER dataset [TKSDM03] and subsequently trained the same model on a variety of domains such as AI, music, politics or science. They used only 100-200 training samples for training on the final domain, to simulate conditions when using such models in real life use-cases, where not a lot of data is available. Their target domains had substantially less overlap in terms of entity types and vocabulary than previous works in the field of cross-domain NER. Furthermore, the majority of their entity types in the target domain are so called domain-specialized entities, like "astronomical objects" or "algorithm" for the domains of natural science or AI. They found that the prediction performance for many entity types, which were not part of the CoNLL dataset, increased as well. However, in our experiments we could only reach a very small improvement by training our model previously on the NuToT dataset before training on our drug dataset. The results from those experiments can be seen in table 4.2.

Since we couldn't gain similar improvements as CrossNER, we figured that we need to stick as close as possible to the original experimental design. We evaluated our dataset, DreamDrug, in the same few-shot scenario as [LXY+20], only 100 samples were used

as training dataset. Additionally we used only the language model BERT-Base-Cased (see section 2.2) from the CrossNER paper, even though the RoBERTa-Base model (see section 2.2) achieved slightly better results. We found it to be more consistent for further analysis, to stick to the original language model used in the work of [LXY⁺20]. The results from these experiments can be found in table 4.3.

ID	Pre-Training Dataset	LM	dapt	dropout	F1	Precision	Recall
1	None	BERT	None	0.0	66.69	68.03	65.41
2	None	BERT	None	0.5	65.40	65.65	65.14
3	None	BERT	All	0.5	71.37	73.48	69.38
4	CoNLL	BERT	All	0.5	66.73	72.44	61.86
5	BTC	BERT	All	0.5	69.97	77.06	64.07
6	WNUT	BERT	All	0.5	67.6	71.69	63.94
7	NuToT	BERT	All	0.5	70.28	73.94	66.97
8	Wiki	BERT	All	0.5	60.53	60.66	60.41

Table 4.3: Few-Shot: Results after training on a different NER dataset in advance. Bold font marks the best model.

When analyzing the results from our experiments with our Few-Shot dataset we found that the impact of domain adaptive pre-training on drug related text has a bigger impact on the performance compared to training on the full dataset (see table 4.2 and 4.3). Moreover, we can observe that we couldn't improve our prediction performance by pre-training on different NER datasets. We assume that performance improvements would require more overlap in terms of vocabulary and entity types.

Hierarchical entity type overlaps were already found problematic in the work of [LXY⁺20]. Their model, which uses exactly the same architecture, had for example problems at the decision between entity type "Person" and "Scientist"/"Politician". In our case drugs are part of the entity type "Product" in the NuToT dataset, therefore it might be hard for the model to realize that now all drugs have to be predicted as "drug" and not as "product" anymore, after training on such a small training set.

Table 4.4 indicates that the vocabulary overlap in our case is quite low and the amount of "drug" entities contained in them is even lower. However, the latter factor might not be important in many cases, since only the NuToT and the distantly annotated Wiki dataset (see chapter 3.1.1 and 3.4.1) have related or overlapping entity types. That is the reason why we assume that the only small performance boost observed is when pre-training on the NuToT (+WNUT) dataset. In this scenario we had overlapping vocabulary and actually somehow overlapping entity classes, since drugs are a subtype of the "product" entity type. However, the results were quite volatile due to the small dataset size.

The failure of boosting the models performance from the distantly annotated Wikipedia drug dataset is not surprising. Even though the dataset contains more drug entities, the annotation quality for illicit drugs via the Spotlight API is insufficient. We assume

Corpus	DreamDrug	CoNLL	BTC	WNUT	NuToT	Wiki
#Unique Words	18785	23623	26953	14879	15926	56023
Vocabulary overlap with our dataset (DreamDrug)	100%	14,39%	13.60%	13.08%	14,52%	11.01%
How many of the drug tokens are present	100%	10.10%	11.37%	6.69%	7.06%	22.53%

Table 4.4: The first row presents the number of individual tokens per dataset. Row number two shows the vocabulary overlap between our corpus and the pre-training NER datasets used. Overlap is defined as the amount of unique words contained in both datasets divided by the total amount of unique words from both datasets. The last row contains the percentage of tokens marked as drug in our dataset contained in the pre-training NER datasets.

that the entity linkage is already a source of error in our distant supervision process (see section 3.4.1). During an exploration of our Wikipedia dataset we found that not a lot of drugs were found via the spotlight API and only exact matches are considered for linkage. This could be probably tackled by data cleaning techniques in the future. However, for future works we recommend the further use of a custom ontology for drug detection like [Sha17], which comprises nearly 30.000 drugs.

The Final Results achieved on the test dataset from the most important models can be observed in table 4.5 and 4.6 for the Few-Shot scenario and trained on the full training dataset respectively. We observed that the best model achieved a precision of 83.70% and a recall of 84,45%. This means 83.70% of all drugs predicted as drug are actually a drug according to our ground truth and that our model found 84,45% of all spans annotated in our ground truth.

Few-Shot Setting			
Model	F1-Score	Precision	Recall
Best FLAIR Model	60.36	57.77	63.19
Our best model WITHOUT training on different NER Task	73.70	74,10	73.30
Our best model WITH training on different NER Task	71.62	74,14	69.27

Table 4.5: Final Results of the most important models in a Few-Shot setting evaluated on the test set

4.5 Prediction Pattern Evaluation

We conducted a qualitative evaluation of our models by using two custom web applications. The first app provided predictions for arbitrary input texts using a model of our choice and the second one enabled us to explore the predictions of the test dataset. The second app also provided means to compare the predictions of two different models for the same

4. EXPERIMENTAL EVALUATION

Full Training Dataset			
Model	F1-Score	Precision	Recall
Best FLAIR Model	72.55	77.39	68.28
Our best model WITHOUT training on different NER Task	84,08	83.70	84,45
Our best model WITH training on different NER Task	83.89	85.55	82.30

Table 4.6: Final Results of the most important models using the full training dataset evaluated on the test set

text and allows the user to filter for different kinds of error e.g. show all texts where False Positives were found. Figure 4.1 shows our comparative WebApp.

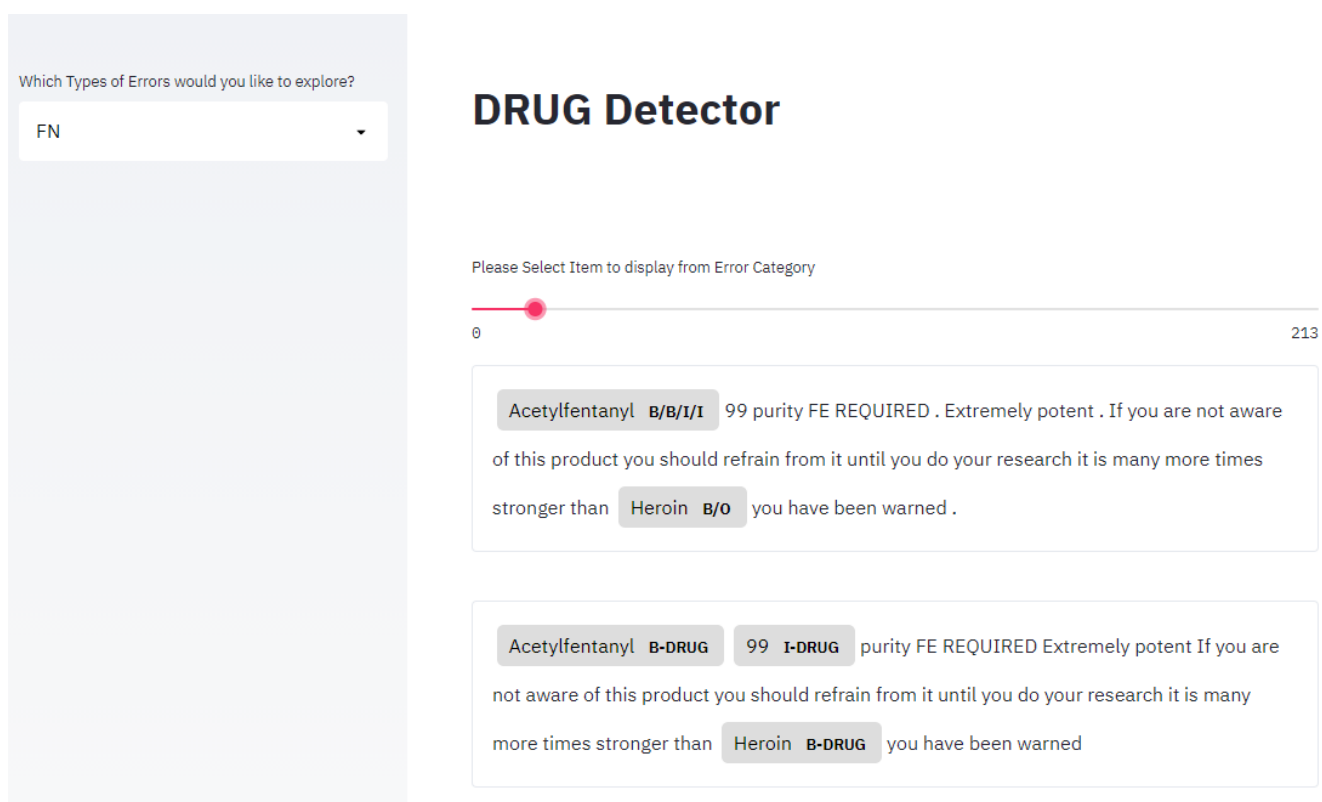


Figure 4.1: Screenshot of the Streamlit WebApp for comparing all predictions where False Negatives were found.

We observed multiple patterns in our qualitative analysis:

- In lists of drugs everything was labelled as "drug". E.g. When a dealer mentioned all cannabis strains he was currently trying to sell or all other drugs he sells, we found that our model marked all spans as drugs, even though some of them would

not be recognized as a drug on its own or maybe aren't even a drug according to our definition.

- If a drug listing started with a word, unknown to our language model it was marked as a drug. For example the token "uantity", actually the misspelled word "Quantity", in "uantity . 100- 100mg sildenafil generic viagra..." was marked as drug. The reason for this is most probably that often unknown tokens in the beginning are actually the drug names and therefore our models assumes that this is a new drug name as well.
- Misspellings were recognized in some cases. For example the term "Cannibis" was recognized as drug, but "weed" was not.

Conclusion and Future Work

In this thesis we shined light onto the usage of transfer learning techniques for Darknet market texts (item listings). We created various text copora for domain adaption (see section 2.1) of language models and evaluated possible performance improvements by task adaptation (see section 2.1 - i.e. pre-training on various other NER datasets). All of those experiments were bench-marked using our newly created illicit drug dataset annotated via crowd-sourcing (see section 3.2). Moreover, we compared our models to an "off-the-shelf" NER model to measure the performance increase by using our methods, which required more effort. This section elaborates on possible conclusions from our experiments with regard to the research questions. We present an answer for each individual question below and continue with a summarization of general results.

5.1 Research Questions

5.1.1 Research Question 1

When using models trained on a different NER task before fine-tuning the model on our drug detection NER task, will they achieve a higher performance in terms of F1-score, if the textual structure of the initial domain is similar to the text structure of the target domain?

Without any pre-training our model achieved 82.17 points in F1-Score. When pre-trained on the NuToT dataset, our model increase by 0.4 points in F1-Score (see Table 4.2). Pre-training on all other datasets reduced the performance. Moreover, in the Few-Shot setting, where only 100 examples are used for training, the best model was not pre-trained on another NER dataset. All pre-training efforts on different NER datasets decreased the performance in this setting (see Table 4.3).

We found that pre-training on a source domain with similar vocabulary and entity types can slightly boost the NER prediction performance. However, in general we cannot

validate the hypothesis that the text structure itself is an important factor, rather the aforementioned vocabulary and entity types in general. Pre-training our model on the NER Task from the Broad Twitter Corpus [DBR16] and W-NUT [DNEL17] (see section 3.1.1) with similarly noisy texts did not help in our final target domain. We assume that the boost in performance in the work of [LXY⁺20] stems from the, despite reduced compared to previous works but still remaining, fraction of overlapping entity types and vocabulary of source and target domain. Moreover, it should be noted that when we achieved a competitive edge by pre-training on NuToT, the improvements were usually so small that slight changes (e.g. changing the seed) could already oppose the results. To sum up, we found no evidence to support this hypothesis in our research.

5.1.2 Research Question 2

Can language model pre-training on well-structured text corpora increase the F1-score of our NER model, even though the target domain is different?

We found that fine-tuning our language model on all provided DAPT text corpora (see section 3.4.2) consistently increased the prediction performance. The results from our big hyper-parameter tuning experiments (see Table 4.1) showed that the drug related Wikipedia articles served well as text corpora for the domain adaptive pre-training. Only domain adaptive pre-training / language model fine-tuning improved the prediction performance by nearly 4 points in F1-Score in the Few-Shot setting. These experimental results provide evidence to support our hypothesis that models working with noisy user generated text can still gain prediction performance by pre-training on well structured domain-related text corpora, as long as their vocabulary overlaps.

5.1.3 Research Question 3

Are distantly supervised datasets such as Wikipedia articles aligned with a Knowledge Graph (DBPedia) able to further boost the F1-score of a NER system with specialized vocabulary and noisy texts?

We couldn't improve the models performance with our distantly supervised dataset and therefore cannot provide any evidence for this hypothesis. Pre-training our model on the distantly annotated Wikipedia drug dataset decreased the models performance by 4 points in F1-Score in case of training on the full dataset and by over 10 points in case of the Few-Shot setting (see Table 4.2 and 4.3). The annotation quality of our distantly annotated dataset was not sufficient and added more noise to the model than improvement. However, we assume that with a high quality distantly annotated dataset e.g. in domains which are well represented in an available Knowledge Graph, a performance boost is likely. We have to leave this question open for further research in the field of transfer learning techniques using semantic technologies.

5.1.4 Research Question 4

Do our custom NER models show a competitive edge in terms of F1-score compared to off-the-shelf models such as the one from Akbik et. Al 2019 [ABV19, ABB⁺19] also known as FLAIR?

When comparing the results from our best models with the results from FLAIR in table 4.5 and 4.6, one can observe that our models outperform those FLAIR models by far. To be specific our models which use embeddings from the same BERT architecture without domain adaptive pre-training (see table 4.1) achieve more than 10 points improvement in terms of F1-Score. Our best model even achieved nearly 12 points more in F1-Score. We can therefore validate this hypothesis and state that training a custom NER model outperformed the off-the-shelf solution that we used for comparison.

5.2 General Results

We are glad to contribute our dataset called DreamDrug for drug focused Named Entity Recognition in the Darknet Market domain. We hope it will act as incubator for further research and ease the creation of subsequent datasets. For access to the dataset, please contact the author via email¹ and provide information about your research interests and institutional affiliation.

We would like to encourage researchers to continue the work on this dataset, by refining our drug annotations, classifying the drug categories for disambiguation use-cases or enhancing it with entity types to provide a wider spectrum of entities in the Darknet domain. This could be done by annotating more data from different categories of the DreamMarket dataset [DZE⁺18]. Another idea for further development would be to enhance the dataset by semantic information about relations. E.g. define which drug is described by the strength description. More precisely a "drug-quantity" or a "drug-strength" relation.

¹DreamDrugDataset@gmail.com

Bibliography

- [ABB⁺19] A. Akbik, T. Bergmann, Duncan Blythe, K. Rasul, Stefan Schweter, and Roland Vollgraf. Flair: An easy-to-use framework for state-of-the-art nlp. In *NAACL-HLT*, 2019.
- [ABK⁺07] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. Dbpedia: A nucleus for a web of open data. In *Proceedings of the 6th International The Semantic Web and 2nd Asian Conference on Asian Semantic Web Conference, ISWC'07/ASWC'07*, page 722–735, Berlin, Heidelberg, 2007. Springer-Verlag.
- [ABV19] A. Akbik, T. Bergmann, and Roland Vollgraf. Pooled contextualized embeddings for named entity recognition. In *NAACL-HLT*, 2019.
- [ACE08] Automatic content extraction 2008 evaluation plan (ace 08) assessment of detection and recognition of entities and relations within and across documents 1. 2008.
- [AMLMS17] Gustavo Aguilar, Suraj Maharjan, A. P. López-Monroy, and T. Solorio. A multi-task approach for named entity recognition in social media data. In *NUT@EMNLP*, 2017.
- [ASM14] Noushin Rezapour Asheghi, S. Sharoff, and K. Markert. Designing and evaluating a reliable corpus of web genres via crowd-sourcing. In *LREC*, 2014.
- [BBR14] Darina Benikova, Chris Biemann, and M. Reznicek. Nosta-d named entity annotation for german: Guidelines and dataset. In *LREC*, 2014.
- [BCDH⁺15] Gwern Branwen, Nicolas Christin, David Décary-Hétu, Rasmus Munksgaard Andersen, StExo, El Presidente, Anonymous, Daryl Lau, Delyan Kratunov Sohlz, Vince Cakic, Van Buskirk, Whom, Michael McKenna, and Sigi Goode. Dark net market archives, 2011-2015. <https://www.gwern.net/DNM-archives>, July 2015. Accessed: 25.01.2021.
- [BDR17] Kalina Bontcheva, Leon Derczynski, and I. Roberts. Crowdsourcing named entity recognition and entity linking corpora. 2017.

- [BVWL20] Alex Brandsen, S. Verberne, M. Wansleeben, and K. Lambers. Creating a dataset for named entity recognition in the archaeology domain. In *LREC*, 2020.
- [DBR16] Leon Derczynski, Kalina Bontcheva, and I. Roberts. Broad twitter corpus: A diverse named entity recognition resource. In *COLING*, 2016.
- [DCLT18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.
- [DLL⁺12] Louise Deléger, Q. Li, T. Lingren, M. Kaiser, Katalin Molnár, Laura Stoutenborough, M. Kouril, K. Marsolo, and I. Solti. Building gold standard corpora for medical natural language processing tasks. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2012:144–53, 2012.
- [DMP⁺04] G. Doddington, A. Mitchell, Mark A. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. The automatic content extraction (ace) program - tasks, data, and evaluation. In *LREC*, 2004.
- [DNEL17] Leon Derczynski, Eric Nichols, M. Erp, and Nut Limsopatham. Results of the wnut2017 shared task on novel and emerging entity recognition. In *NUT@EMNLP*, 2017.
- [DZE⁺18] Po-Yi Du, N. Zhang, Mohammedreza Ebrahimi, Sagar Samtani, Ben Lazarine, N. Arnold, R. Dunn, Sandeep Suntwal, Guadalupe Angeles, Robert Schweitzer, and H. Chen. Identifying, collecting, and presenting hacker community data: Forums, irc, carding shops, and dnms. *2018 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pages 70–75, 2018.
- [EBNA16] K. J. Espinosa, R. Batista-Navarro, and S. Ananiadou. Learning to recognise named entities in tweets by exploiting weakly labelled data. In *NUT@COLING*, 2016.
- [FA21] U.S. Food and Drug Administration. Drug definition by u.s. food and drug administration. <https://www.fda.gov/industry/regulated-products/human-drugs>, 2021.
- [FEN09] K. Fort, Maud Ehrmann, and A. Nazarenko. Towards a methodology for named entities annotation. In *Linguistic Annotation Workshop*, 2009.
- [FGM05] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 363–370, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.

- [FLK19] Alexander Fritzier, V. Logacheva, and M. Kretov. Few-shot classification in named entity recognition task. *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, 2019.
- [FLRS⁺15] O. Feyisetan, Markus Luczak-Rösch, E. Simperl, Ramine Tinati, and N. Shadbolt. Towards hybrid ner: A study of content and crowdsourcing-related performance factors. In *ESWC*, 2015.
- [FMK⁺10] Timothy W. Finin, William Murnane, A. Karandikar, Nicholas Keller, Justin Martineau, and Mark Dredze. Annotating named entities in twitter data with crowdsourcing. In *Mturk@HLT-NAACL*, 2010.
- [FSLR⁺18] O. Feyisetan, E. Simperl, Markus Luczak-Rösch, Ramine Tinati, and N. Shadbolt. An extended study of content and crowdsourcing-related performance factors in named entity annotation. *Semantic Web*, 9:355–379, 2018.
- [GCRF20] Inês Gomes, Rui Correia, Jorge Ribeiro, and João Freitas. Effort estimation in named entity tagging tasks. In *LREC*, 2020.
- [GGK18] Archana Goyal, Vishal Gupta, and M. Kumar. Recent named entity recognition and classification techniques: A systematic review. *Comput. Sci. Rev.*, 29:21–43, 2018.
- [GS96] R. Grishman and B. Sundheim. Message understanding conference- 6: A brief history. In *COLING*, 1996.
- [HKGNH18] Maximilian Hofer, A. Kormilitzin, Paul Goldberg, and A. Nevado-Holgado. Few-shot learning for named entity recognition in medical text. *ArXiv*, abs/1811.05468, 2018.
- [HLS⁺20] Jiaxin Huang, C. Li, Krishan Subudhi, D. Jose, S. Balakrishnan, W. Chen, Baolin Peng, Jianfeng Gao, and Jiawei Han. Few-shot named entity recognition: A comprehensive study. *ArXiv*, abs/2012.14978, 2020.
- [HMP⁺06] Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. Ontonotes: The 90 In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, NAACL-Short '06, page 57–60, USA, 2006. Association for Computational Linguistics.
- [HMYLB20] Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spaCy: Industrial-strength Natural Language Processing in Python, 2020.
- [hom11] E. P. hommeaux. Sparql query language for rdf. 2011.

- [HR05] G. Hripcsak and A. Rothschild. Technical brief: Agreement, the f-measure, and reliability in information retrieval. *Journal of the American Medical Informatics Association : JAMIA*, 12 3:296–8, 2005.
- [JMGB20] Mona Jalal, Kate K. Mays, L. Guo, and Margrit Betke. Performance comparison of crowdworkers and nlp tools on named-entity recognition and sentiment analysis of political tweets. *ArXiv*, abs/2002.04181, 2020.
- [JXZ19] Chen Jia, Liang Xiao, and Y. Zhang. Cross-domain ner using cross-domain language modeling. In *ACL*, 2019.
- [JZ20] Chen Jia and Y. Zhang. Multi-cell compositional lstm for ner domain adaptation. In *ACL*, 2020.
- [LCFW20] Jing Li, Billy Chiu, Shanshan Feng, and H. Wang. Few-shot named entity recognition via meta-learning. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1, 2020.
- [LEPY10] Nolan Lawson, Kevin Eustice, Mike Perkowitz, and Meliha Yetisgen-Yildiz. Annotating large email datasets for named entity recognition with mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk*, CSLDAMT ’10, page 71–79, USA, 2010. Association for Computational Linguistics.
- [LMP01] J. Lafferty, A. McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001.
- [LNC⁺18] Di Lu, Leonardo Neves, V. Carvalho, N. Zhang, and Heng Ji. Visual attention model for name tagging in multimodal social media. In *ACL*, 2018.
- [LOG⁺19] Y. Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692, 2019.
- [LXY⁺20] Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. Crossner: Evaluating cross-domain named entity recognition. 2020.
- [MCCD13] Tomas Mikolov, Kai Chen, G. S. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In *ICLR*, 2013.
- [MFP00] A. McCallum, D. Freitag, and Fernando C Pereira. Maximum entropy markov models for information extraction and segmentation. In *ICML*, 2000.

- [NBB⁺06] C. Nédellec, P. Bessières, R. Bossy, A. Kotoujansky, and Alain-Pierre Manine. Annotation guidelines for machine learning-based named entity recognition in microbiology. 2006.
- [NFAFR20] Mhd Wesam Al Nabki, E. Fidalgo, E. Alegre, and Laura Fernández-Robles. Improving named entity recognition in noisy user-generated text with local distance neighbor feature. *Neurocomputing*, 382:1–11, 2020.
- [NFM19] Mhd Wesam Al Nabki, E. Fernandez, and Javier Velasco Mata. Darkner: a platform for named entity recognition in tor darknet. 2019.
- [NRR⁺13] Joel Nothman, Nicky Ringland, Will Radford, Tara Murphy, and James R. Curran. Learning multilingual named entity recognition from wikipedia. *Artificial Intelligence*, 194:151–175, 2013. Artificial Intelligence, Wikipedia and Semi-Structured Resources.
- [QZZ⁺20] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2020.
- [SACM19] Alisa Smirnova, J. Audiffren, and P. Cudré-Mauroux. Distant supervision from knowledge graphs. In *Encyclopedia of Big Data Technologies*, 2019.
- [SBDS14] M. Sabou, Kalina Bontcheva, Leon Derczynski, and A. Scharl. Corpus annotation through crowdsourcing: Towards best practice guidelines. In *LREC*, 2014.
- [Sha17] Mark E. Sharp. Toward a comprehensive drug ontology: extraction of drug-indication relations from diverse information sources. *Journal of Biomedical Semantics*, 8, 2017.
- [Shu10] Nakatani Shuyo. Language detection library for java, 2010.
- [SSZ17] J. Snell, Kevin Swersky, and R. Zemel. Prototypical networks for few-shot learning. In *NIPS*, 2017.
- [TKSDM03] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147, 2003.
- [TMS⁺21] Maxim Tkachenko, Mikhail Malyuk, Nikita Shevchenko, Andrey Holmanyuk, and Nikolai Liubimov. Label Studio: Data labeling software, 2020-2021. Open source software available from <https://github.com/heartexlabs/label-studio>.

- [VG05] A. Viera and J. Garrett. Understanding interobserver agreement: the kappa statistic. *Family medicine*, 37 5:360–3, 2005.
- [VSP⁺17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.
- [WDS⁺20] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics.
- [YK20a] Y. Yang and Arzoo Katiyar. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. *ArXiv*, abs/2010.02405, 2020.
- [YK20b] Y. Yang and Arzoo Katiyar. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. *ArXiv*, abs/2010.02405, 2020.

Appendix

A.1 Project Execution - Detailed

The following sections describe how we executed the data annotation project on the two platforms Appen and Amazon Mechanical Turk. We state relevant parameters for quality control and our quality improvement measures. Nonetheless, those will not ensure Reproducibility in future projects and need to be interpreted as experiences rather than scientific results.

A.1.1 Crowd-sourcing Project Execution - Appen

Before starting a crowd-sourcing campaign one has to decide upon certain settings provided by Appen. The exact parameters can be found in table A.1 for the different campaigns. After every campaign we incorporated the results into a refined set of instructions and test questions to improve annotation quality. Moreover, we used the geo-restriction possibility of Appen, so all participating crowd-workers were based in the United States and United Kingdom. We, hypothesize that those users are more likely to resemble typical customers and vendors of Darknet drug listings, since items are usually not shipped to different continents due to detection probability at customs.

Each crowd-worker had to qualify for this project by initially annotating examples. If the worker reaches a score above the Minimum Accuracy presented in table A.1 he is accepted for the job. During the ongoing crowd-sourcing campaign annotators had to maintain this high level of agreement with our gold annotations. Overall, 20% of all questions were set to be test questions. This measure ensures a high quality of annotations and prevents annotators from "cheating" or randomly annotating tokens.

campaign ID	1	3	4	5	6
#Rows	100	100	150	250	300
% Test Questions	20%	20%	20%	20%	20%
Payment per Row	0.08\$	0.08\$	0.08\$	0.08\$	0.06\$
Annotator Competence	Level 3	Level 3	Level 3	Level 3	Level 3
Minimum Accuracy	75-80%	75%	75-80%	75%	75%
Judgements Per Row	3-static	3-static	3-static	5 - dyn.	3-static
Trusted Judgements	450	322	665	1672	909
Untrusted Judgements	115	234	121	419	273
Total Cost	52\$	61\$	91\$	238\$	106\$
Annotators total trusted/untrusted/declined	52/18/154	32/30/119	51/59/205	101/102/299	113/34/111
Cohen's Kappa	0.57	0.35	0.43	0.39	0.43
Pairwise F1-Score (makro/mikro)	0.68/0.66	0.60/0.54	0.62/0.59	0.62/0.62	0.55/0.59

Table A.1: Parameters and results of Appen crowdsourcing campaigns.

Inter Annotation Agreement - Metric

As quality measure for our annotations we did not solely rely on the Inter Annotator Agreement via Cohen's Kappa due to the recommendation of [BVWL20, DLL⁺12]. According to [DLL⁺12] Cohen's Kappa requires the number of negative cases, which is not known in case of NER. Furthermore, due to the imbalance between "O" (other) and "drug" entity class, a general imbalance in Named Entity Annotation tasks, and the instability of Cohen's Kappa we additionally report the averaged pairwise agreement F1 Score between annotators. [HR05] shows that Cohen's Kappa approaches the pairwise F1-Score, if a sufficient amount of negative samples is present. Otherwise, Cohen's Kappa can be unreliable. We report this measure solely for comparative reasons, since it's still widely used in the literature. In small batches such as run 1 and 3 in table A.1 one can observe the volatile behaviour of Cohen's Kappa. The decrease in F-Score is 8-12 points but Cohen's Kappa went down by 22 points.

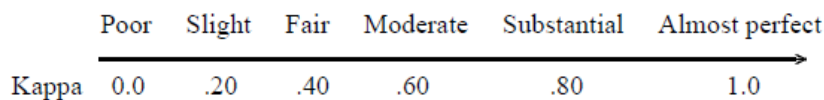


Figure A.1: Agreement according to Cohen's kappa from [VG05]

Batch 1

Batch 1 showed promising results, especially for our initial test set-up. The results achieved, in terms of F-Score, were comparable and sometimes higher than a comparative

performance study about named entity annotations from crowd-workers [FSLR⁺18]. This study conducted named entity annotation of less complex classes like Person, Location or Organization on twitter datasets and measured the IAA in different annotation settings. Their results can be found in Table A.2.

Dataset	Person	Organization	Location	Misc.
Finin Dataset	63.34 / 46.33	36.00 / 35.54	63.31 / 53.43	-
Ritter Dataset	52.98 / 43.65	33.30 / 33.55	57.33 / 56.31	20.13 / 18.91
MSM 2013	86.91 / 79.46	40.90 / 44,55	63.75 / 53.94	15.37 / 9.98

Table A.2: Performance of Crowdworkers IAA measured using Cohen' Kappa in [FSLR⁺18]. The two values are reported for different annotation conditions to evaluate performance impact factors.

Since we experienced, that crowd-workers struggled with comprehending the annotation guidelines we decided to remove verbal descriptions as much as possible and replace all examples with screenshots of annotated examples to enable a faster understanding with less attention needed. The remaining verbal descriptions were simplified as much as possible including the usage of plain language with only one thought per sentence, simple sentence structure, no advanced vocabulary, active language and always accompanied by an example.

Batch 2

In Batch 2 we experienced difficulties with the Appen platform. The annotation ontology required all users to annotate at least a single entity, even though no drug entity was present. The option to skip annotations, was only available to the Admin user. These issues resulted in an inconsistent job status and made us cancel Batch 2. Therefore Batch 2 is not present in table A.1.

However, the small subset of valid annotations in Batch 2 showed us that we cannot improve the set of rules to cover corner cases. E.g. the explicit description that pill imprints shall or shall not be labelled seems to be prevented by the ambiguity of our text corpus. We assume it is very hard for annotators to comprehend if a term is a unique pill imprint, a pill identification number (not necessarily a pill imprint) or just another amount of active agent or purity.

The issues from Batch 2 required us to create another job set-up in run 3, which included a "None" option and further improvements learned from run 1 and 2. This included:

- That noisy characters shall be enclosed in the Annotation e.g. the span "White . Widow" is considered a drug as a whole and shall not be labelled as the tokens "White" and "Widow", which cannot be identified as such on their own.

- Pill Imprints are no longer explicitly stated to be labelled as a drug. This information is too complex to be extracted by crowd workers since this knowledge can be presented in various forms and can easily be mistaken for a drug name.
- Processed edibles shall not be labelled as a drug, since these usually refer to ordinary food e.g. "Megabelt Chocolate Bar" does not refer to a specific drug.
- That we reduced the minimum accuracy threshold for test questions. This should reduce the amount of workers which are passing the initial test by chance, but drop out later since they cannot maintain the accuracy requirements. However, this seemed to have shown no measurable impact.
- Added a rule to encourage, labelling drug classes such as "Opioids" or "Benzodiazepines" as drug.
- Improved the optical appearance of rules e.g. Headline Font instead of bullet-point for each rule.
- Added a "None" class to the annotation ontology, since the workers have to annotate at least one token.
- A new set of test questions which were reviewed twice before the start.

Batch 3

Unfortunately, these improvements did not result in a better F-Score for Batch 3 compared to Batch 1. We hypothesize, that this was caused by the high amount of easy test questions in Run3. So annotators who didn't read or understand the annotation guidelines were able to enter the Task, but were removed from the job later on due to their long term performance. There was a high amount of applicants for each job and only 7 test questions to check their initial performance. A high amount of easy test questions resulted in a high amount of workers passing the entry barrier, just by chance. However, those were removed in the long run, since they couldn't maintain a high accuracy on continuous test questions. This resulted in increased costs, because we pay all annotations no matter if they are valid or invalid. Moreover, it seems possible that the initial Run 1 achieved a very high accuracy by chance due to the small sample size.

Batch 4

For our first bigger sample with Run 4 we decided to implement a few improvements:

- Minor description improvements by changing formulations.
- Raising the test question accuracy up to 80%.
- Add more difficult test questions and remove a few "too simple" ones.

Due to the rapid increase of invalid annotations (we have to pay for annotations which become untrusted later on as well) we decreased the accuracy back down to 75% again

during Batch 4. This caused the overhead of valid annotations. This is needed, since otherwise our budget would not cover the needed amount of annotations.

Batch 5

In Batch 5 we changed to a dynamic setting for evaluating how many annotations we need (2-5 based on the confidence scores of the annotators which are calculated based on the test question performance). This resulted in a steep increase of costs (cost details can be found in table A.1). We spent more money, since the agreement of annotators was usually below the trust threshold of 80%. Therefore we decided that we will stay with 3 static annotations per row. It shall be noted that a higher amount of annotations per row did not increase annotation accuracy or pairwise agreement.

Batch 6

Due to the particularly high income of many annotators in Batch 5 combined with no increase in quality or effort they spend on the task we tried a lower payment in Batch 6. Far less people applied for the project compared to earlier runs. Due to the high reward of our earlier jobs, compared with a cost summary of other projects in [SBDS14], we assume our jobs were probably some of the best paid jobs at this time. We only spend one third of the costs of Batch 5 per row (0.95\$ vs 0.35\$) for Batch 6. Which stems partially from the fact that in earlier batches all annotators tried to work on this task due to its high payment, even though they weren't able to grasp the instructions and their test-question accuracy fell below the threshold. Therefore, these already payed annotations became invalid and had to be re-done.

In this batch it seems that the annotators understood the task quite well and we ended up having far less invalid judgements, compared to batch 5. Even though the people spend a significantly smaller amount of time per annotation we observed a reduction of only 3-5 points F1-Score (see table A.1) in terms of pairwise agreement. Since we review the annotations afterwards anyways, it makes more sense to us to annotate a bigger amount of rows.

Unfortunately we weren't able to continue our annotation project from here on with Appen. Even though we managed to find a good set-up for starting to annotate bigger amounts of data, Appen would have required us to pay license fees in addition to the crowd payments and Appen's 20% add-up, which was not feasible.

A.1.2 Project Execution - Amazon Mechanical Turk

Due to the license problems with Appen, we decided to switch to the Amazon Mechanical Turk platform in order to continue our crowd-sourcing project. Initially we tried to re-use exactly the same annotation guidelines from Appen for Amazon Mechanical Turk. However, the workflow is quite different on the MTurk platform. Crowd-workers do not have to participate in an initial test to prove their understanding of the task and

there are no test questions to measure the agreement between crowd-workers and a gold standard. They can engage in any Human Intelligence Task (HIT), as MTurk calls an annotation task, if they fulfill the requirements. Possible examples for requirements are geographic location or personal properties, such as gender, age or education.

Working with MTurk causes substantially more work on the side of the requester / ordering party, compared to Appen, since annotators need to be monitored manually and the requester has to decide whether you accept a conducted HIT (and pay) or decline it. The only accuracy requirement we can use is the HIT approval rate for all of the requester's HITs, which would remove annotators falling below the threshold. We set this threshold to 75%. Equally to the Appen set-up only crowd-workers from the United States and the United Kingdom were allowed to participate. Furthermore, we required our annotators to have the "Master" qualification, which we considered equal to "stage 3" annotator from Appen. During the project execution we found out that the quality requirements for "Master" annotators are higher, but quite in-transparent.

The only description of those requirements we found from Amazon was - "Mechanical Turk has built technology which analyzes worker performance, identifies high performing Workers, and monitors their performance over time. Workers who have demonstrated excellence across a wide range of tasks are awarded the masters qualification. Masters must continue to pass our statistical monitoring to retain the Mechanical Turks masters qualification".

There are also improvements when working with MTurk. One feature is, that due to the bonus payments and stronger quality controls the relation between purchaser/requester and crowd-workers is closer. We received valuable feedback from MTurks crowd-workers and were able to reward good performance via bonus payments, in order to encourage further performance improvements as recommended in [LEPY10].

Encoding Problems

It should be noted that we experienced issues when trying to re-use the same file format. Amazons MTurk platform doesn't interpret quoted character sequences as strings, which causes delimiter problems and other parsing errors when uploading .csv files. Unfortunately, at the time being there is only the option to upload .csv files, so we were forced to alter our pre-processing for this platform.

Batch 1 & 2

In our initial tests we experienced that crowd-workers on MTurk, would not engage for the same reward per HIT/task as in Appen and would not continue to work for long on a task with such long guidelines. Therefore, we provided crowd-workers a refined small task description, used for the sidebar earlier on. The long description was only available if they check the "more Instructions" page.

The Inter-Annotator Agreement of MTurk crowd-workers was significantly higher compared to Appen (see table A.1 and A.3). We assume that this was caused by the closer relationship from requester to crowd-workers with the "Master" qualification. We only hired "Master" annotators which cost approx. 0.02-0.03\$ more per HIT, but Amazon ensured that those workers have very high approval scores by the requesters and therefore usually satisfy the quality requirements.

campaign ID	1	2	3	4	5	6	7	8	9
Text Type	short	short	short	Long	Long	Long	Long	short	short
#Rows	100	300	600	100	200	200	200	500	300
Reward per Row	0.11\$	0.11\$	0.11\$	0.25\$	0.25\$	0.30\$	0.25\$	0.09\$	0.11\$
Minimum HIT Acceptance	75%	75%	75%	75%	75%	75%	75%	75%	75%
Annot. Per Row	3	3	3	3	3	3	3	3	3
Trusted Judgements	300	870	1787	278	583	600	598	1498	898
Untrusted Judgements	0	28	151	13	0	0	0	6	0
Total Cost	42\$	126\$	252\$	93\$	186\$	228\$	186\$	165\$	126\$
Cohen's Kappa	0.81	0.74	0.85	0.78	0.79	0.69	0.72	0.72	0.78
Pairwise F1 Makro	0.83	0.77	0.86	0.79	0.80	0.70	0.73	0.74	0.79
Pairwise F1 Mikro	0.82	0.77	0.81	0.80	0.79	0.81	0.79	0.74	0.76

Table A.3: Parameters and results of MTurk crowdsourcing campaigns.

It should be noted that the measures for Cohen's kappa and F1 agreement scores are calculated character-wise in table A.3, compared to the token based calculation earlier on with Appen in table A.1. This difference is caused by the character/token based text handling of the different platforms.

Batch 3:

In our third run on MTurk we executed an even bigger batch, which attracted a bigger pool of annotators. This did not negatively impact the IAA, but increased the review efforts on our side significantly, since we needed to ensure the annotation quality for each annotator at least on a sample basis. This is needed in the beginning, since the IAA will not ultimately ensure a proper annotation quality, just an agreement between crowd-workers.

During these initial batches we established trust in well performing annotators, so we can efficiently measure future performances based on the IAA with those trusted annotators. From here on we still checked data samples, but mostly relied on the IAA with trusted annotators, where we already ensured the annotation quality.

The continuous quality assurance process was necessary to improve the annotation of corner cases and to mitigate the participation of annotators who weren't able to comprehend the task instructions or just didn't read them at all. Moreover, we found two cases where annotators did not commit any annotation at all. Even though their annotation time per sample was similar to other annotators, they committed mostly empty annotations.

Batch 4:

In our fourth batch we started annotating long texts. Those texts had at least 1000 characters in length and up to 3000 as maximum length. We found that annotators were not highly interested in engaging in this batch and only a few well known and high quality annotators participated. We assumed, that this was caused by the batch having been published on a Friday and the upcoming weekend.

Batch 5:

When batch 4 was finished a few days after its start, the subsequent batch 5 was not receiving a lot of attention by the crowd-workers as well. Due to a hint by a dedicated crowd-worker we found out that the lack in participation was caused by our "bad" approval rate, which was at about 95% at this time and a bad review about our tasks in the biggest MTurk forum ¹.

Even though this is not mentioned in the relevant academic literature, the approval rate of a requester highly impacts the willingness of "Master" crowd-workers to participate in his/her tasks. We discovered that crowd-workers would only engage with requesters with a good reputation in forums and a high approval rate of above (crowd-workers recommended at least 98%). This might seem contradictory to MTurks task set-up where one can decide upon accepting/declining HITs. One solution for this problem might be the current emergence of "entry tests" similar to Appen. This measure is supposed to prevent unqualified workers from participating and therefore preventing regular declination of their HITs or blocking users. Higher declination rates and blocks can result in the loss of the "Master" qualification and therefore in the loss of substantial parts of income. This is the reason why, "Master" annotators are unlikely to engage in task for users with low acceptance rates. Unfortunately, we did not have the time to establish such an entry test, since this is only possible via the AWS API and would have caused extensive efforts.

We were able to resolve the conflict with empty annotations from Batch 4, which caused the bad review in the first place. This ultimately resulted in a more positive edit of the review which improved our reputation in the MTurk Forum. Moreover, we posthum accepted the majority of invalid HITs to boost our approval rate. Even though this cost 15\$ and we didn't use the data for further processing, the higher acceptance rate was worth the investment.

¹<https://turkerview.com/requesters/ARC1S630YUZZE/reviews> - last accessed 05.05.2021

Moreover, the crowd-worker gave us further hints to increase the workers performance by drastically increasing the maximum assignment time of HITs from 3 minutes to a long time span, so they can accept a lot of tasks in a batch at once and then take their time to work on them relaxed. We decided to increase the maximum assignment time from 3minutes to 45 minutes.

Batch 6 - 9:

With the new set-up and a good reputation our final batches were each finished in less than a day. The only experiment we conducted was a pay reduction in the final test, to see if it will affect performance and it turned out it does. The IAA dropped significantly (see table A.3) and the annotation quality and time, which the annotators took went down. It should be mentioned that our payment was always ethical according to our calculations and according to the reviews even higher than the recommendations (12\$/hr for the "cheap" batch 8 and 13.2-18\$ for the earlier batches).

A.2 Annotation Guidelines

A.2.1 Annotation Guidelines for Experts

In this task you will annotate drugs in item listings on Darknet markets. We will present you the name of the item listing as context and item description as annotation target. We want you to annotate all tokens which represent the concept of a drug. Since the definition of drug is quite vague we explicitly include the following in our definition of a drug.

- legal drugs e.g.:
 - Alcohol
 - Aspirin
- partially legal / prescription drugs e.g.:
 - Cannabis (legal in some states in the US)
 - Xanax
 - Steroids
- illicit drugs e.g.:
 - Cocaine
 - Heroin

Moreover we follow the FDA definition of a drug which says that a drug is:

1. a substance intended for use in the diagnosis, cure, mitigation, treatment, or prevention of disease

2. a substance other than food intended to affect the structure or function of the body

(<https://www.fda.gov/industry/regulated-products/human-drugs>)

We want you to label all drugs which fulfill this definition, especially drugs of the aforementioned categories. The tokens we want you to label shall be clearly connected to the concept of a drug. This connection can be the marketing name of a medical drug, the street name of an illicit drug, or the chemical description of a research drug. When it comes to names for drugs we include all kinds of precise names for drugs, subcategories of drugs or drug identifiers. Possible examples could be:

- "White widow" - is the name of a Cannabis strain and therefore a drug
- "Diazepam" - is the active agent of the well known drug "Valium". Both tokens would resemble a drug for us.
- "Oxys" - the street name for "Oxycodone" represents a drug.

The important key is that the token should uniquely identify a drug or group of drugs. This can be a high-level category of drugs like opiates, cocaine or benzodiazepines. Specific marketing names like Xanax or informal names like the strain of a cannabis sort. Low-level chemical compounds, which clearly represent a drug like "MDMA" or "THC". Street/Slang names which clearly identify a drug like "XTC" for Ecstasy.

Below, we will present guidelines for this annotation process. In our exemplary item descriptions we will label spans which are supposed to be labelled as "Drug" with "<Drug>" in the beginning and "</Drug>" in the end.

1. Any kind of unique name or identifier of a drug is labelled as "Drug". Usually you can infer drug names from the context or name of the item sold.
 - "Silver Super Haze", "Jack Herer" are cannabis strains and therefore drugs.
 - "Indica", "Sativa" and "Hybrid" are categories of marijuana strains and drugs.
 - "Valium" is the marketing name of Diazepam in certain countries and therefore a drug.
 - "Benzodiazepines" or "steroids" refer to whole types of drugs and shall be labelled as such.
 - "1 gram of the finest <Drug>Amnesia Haze</Drug>. One of my own favorites. 1 gram for only 10 euros. You want the BEST of the BEST. Our <Drug>AMNESIA</Drug> is in the offer for the few next weeks" - Its clear

- that "AMNESIA" refers to the cannabis strain "Amnesia Haze" and is therefore a drug.
2. We are only interested in the drug itself. If a drug is surrounded by additional descriptive terms we only label them if they really belong to the drug noun/identifier. Visual descriptions like appearance adjectives (color, form etc.), state (powder, pill, fluid) or strength (e.g. 200mg) are not part of the drug.
 - "10 | Dom Perignon <Drug>XTC</Drug> Pills 200mg (ESCROW)| UK Ven" - Only the token XTC represents the concept of a drug. It is not of interest that it's a pill or that its shape looks like a Dom Perignon logo.
 3. Label ALL drugs in the item description. The name of the item sold (provided as context) will indicate the primary item sold, but there can be other drugs mentioned in the description as well.
 - Name (only for context):
 "10mg roxycodone IR 10 pills for 110\$ ships from US"
 description:
 "10 mg <Drug>roxycodone</Drug> 10 pills for 110 instant release from my script from pharmacy no <Drug>fent</Drug>. The pills in the picture are the ones you will receive different from <Drug>Percocet</Drug> because it has no tylenol so they are better for your liver"
 4. Chemical Details ARE part of the drug.
 - "4 x <Drug>Viagra Sildenafil Citrate</Drug> 100 mg by Cipla. TOP QUALITY. Pharmaceutical. This is legitimate quality <Drug>Viagra </Drug> from Cipla" - Citrate is part of the drug identifier.
 - "1 Gram <Drug>Fentanyl HCL</Drug> very potent stuff AT A SALE PRICES" - HCL (Hydrochloride) is part of the drug.
 5. Abbreviations of Drugs are Drugs.
 - "XTC" is a drug, because its the short form of "Ecstasy"
 - "BTH" is a drug, because its short for Black Tar Heroin. Usually such an example will be clear from context. Either from the "Name" context provided or it will be introduced in the text.
 6. Activate Agents of Drugs shall be labelled as drug
 - "THC", "CBD" or "cannabinoids" are the active agents of Cannabis and therefore a drug

- "Diazepam" is the active agent of "Valium" and therefore a drug.
7. Slang names are only labelled as drugs if they represent a clear reference to a specific drug.
 - "Skunk" will clearly refer to a cannabis strain and shall be labelled.
 - "Oxys" will refer to Oxycodone and shall be labelled.
 - "Fishscale Powder", "These Mitsubishi Pills" or "This strain" will refer to a drug and in this context (in combination with an earlier sentence) may be clear, but will not generally identify which drug it is. Therefore, we do NOT label it.
 8. Mentions of drugs like "Super outdoor buds", "rocks", "flaky powder", "This pill" or "crystals" are occurring in the text. However, we will NOT label them as drugs, since its usually only a co-reference to a drug.
 - "1 LB <Drug>Blackberry Diesel</Drug> - Some high grade indoor bud - Better , cheaper bud than the other vendors" - Even though "indoor bud" refers to the cannabis, it doesn't clearly point to the concept of a drug, since its rather a Co-Reference.
 9. The form is NOT part of the drug. Whether it is a pill, paste, oil or powder is not in scope of this project.
 - "Pure quality champagne crystal <Drug>mdma</Drug> 85 [SEP] sent as rocks or powder" - In this case crystal is not part of the drug.
 - "200ml <Drug>Hash</Drug> oil vaping cartridges" - Only "Hash" is a drug in this example.
 - "This listing is for 50 Grams of the Finest <Drug>Amphetamine</Drug> <Drug>Speed</Drug> Paste with a 79 - 85 Purity directly from the lab in the Netherlands" - As you can see "Paste" is not part of the drug.
 10. The Brand of the drug is not part of the drug
 - e.g. "Pfizer <Drug> Xanax </Drug> 1mg x 20 Pills"
 11. Natural chemicals which are NOT a drug. A neurotransmitter like dopamin or hormones like estrogen are NOT a drug, since its a natural compound of our body and not a created with the intentions of the FDA drug definitions 1 and 2.
 - "<Drug>Exemestane</Drug> is a steroidal Aromatase Inhibitor AI that is most commonly known as <Drug>Aromasin</Drug> [SEP] Aromatase Inhibitor would begin to gain a lot of popularity among anabolic <Drug>steroid</Drug> users for its ability to protect against estrogenic related side effects" - Only nouns which refer to a drug shall be labelled.

- "<Drug>Reductil</Drug> [SEP] Each tab contains 20 mg <Drug>Sibutramine</Drug> [SEP] This listing is for 60 Tabs [SEP] <Drug>Sibutramine</Drug> is a medication that assists with weight - loss by altering neurotransmitters within the brain [SEP] <Drug>Sibutramine</Drug> blocks the reuptake of the neurotransmitters dopamine , norepinephrine , and serotonin"- As you can see in this example "dopamine" or "serotonin" are no drugs.

A.2.2 Annotator Guidelines Sidebar - Appen

This small heuristic was presented to the users on the sidebar whenever, they clicked on the drug entity type info:

A drug is every substance, which is created to change your body or mind. To make you high, to cure illnesses or improve your body! E.g. cannabis strain "Gorilla Glue" or medicine name like Nandralone or Valium.

We want you to label ALL tokens which:

- Clearly represent a specific drug e.g. hash, meth
- Chemical Details which extend the drug e.g. Viagra Sildenafil Citrate
- Chemical Compounds e.g. "3 , 4 - methylphenidate"
- Short names e.g. XTC
- Agents e.g. THC, CBD or Diazepam
- Slang drug names: Coke, Fent,
- Drug categories e.g. Indica/hybrid, benzodiazepines, opiates

DO NOT label tokens like:

- Description/Adjectives like drug strength, colour, form (powder/pill..) or looks etc.
- Vague references like "this pill" or "indoor bud"
- Producer of drugs e.g. Pfizer
- Natural chemicals in your body e.g. estrogen/dopamin

IF you cannot find ANY drug, label the first token as "None"

LABEL tokens of the same drug name together! DO NOT separate parts of a chemical or marijuana name -> Check long description for that!

Annotation Guidelines - Appen

Instructions ^

Overview

In this task you will use the text annotation tool to highlight drugs in texts from the Darknet. This highlighting means you label pieces of text (called "token") in product descriptions of Darknet Markets. The items have a name and a description. We want you to label all DRUGS you can find in the description.

HOW TO DO THIS?

1. Read the Context below the text to get an idea of which items could be contained in the text

genuine cmaxx androlex 250 mg . 10 ml bottle x 2 bottles test e 100 mg test
p 75 mg test deca 25 mg test i so 50 mg .

Context
cmaxx androlex 250mg. 10ml bottle x2 bottles

2. Read the provided drug description and label ALL drugs --> Not just the one in the context.

3. Are you unsure about what is a drug? No problem - Click on the info of the "Drug" Class or re-open this text.



Annotation Guidelines

Label every unique drug name you can find! Typical Drug Examples are:

- Marijuana Strains or types - s.e.g. "Gorilla Glue", "Indica / Sativa / Hybrid", "Super silver Haze" or "Girl Scout Cookies"
- Types of Drugs or Agents e.g. "Cocaine", "Crystal Meth", "MDMA" or "THC"
- Legal or partial legal drugs e.g. "Alcohol", "Testosterone Enthalate", "Aspirin" or "Viagra"
- Slang names e.g. Coke, Meth, mushrooms, hash, weed

Label ONLY the drug - Descriptions or Adjectives are not part of the drug

(like "Colour, Form or If its powder / oil / paste / pill or the strength of a drug e.g. 200mg)

100 XTC tablets containing 230 mg of the finest quality MDMA available .
These grey pills have been lab tested . These beauties are shaped like an actual

Context
XTC 'GREY SIMCARDS' 230MG: 100 pc

We want you to label ALL drugs in the text - Pay attention to label drugs with multiple words as one drug. (Strawberry is not drug without amnesia)

If there is a " " or " ' " contained in the middle of a name label it as well. "Strawberry . Amnesia" or " Strawberry - Amnesia" would also be a drug.

Top shelf coffeeshop quality . A powerful and uplifting flower from Dinafem
 Seeds , ^{Drug} Strawberry Amnesia is a strain made in sativa heaven . Bred from
^{Drug} Strawberry Cough and ^{Drug} Amnesia , this strain delivers the familiar sweet strawberry
 and earthy flavors of its parents . Having the typical energizing and euphoric
 effects of a ^{Drug} sativa , ^{Drug} Strawberry Amnesia also induces the calming body high
 from its distant ^{Drug} indica relatives . The dark green buds of ^{Drug} Strawberry Amnesia
 are very dense and heavily coated in resin , so this potent ^{Drug} sativa should be
 handled with caution .

Context

Strawberry Amnesia - *New strain* - AAAA+ - 3.5G

Chemical details are part of the drug.

- Sildenafil Citrate is a single drug and need to be labelled together (not separate)

^{Drug} Sildenafil Citrate 100 mg ^{Drug} Viagra is a Generic medicine used for Penis erectile

- Drug Chemicals can also be drugs on their own. - AND Include "-" or "'" in the labelling

This listing is for 100 mg of ^{Drug} 5 - MeO - MiPT .

Context

100mg 5-MeO-MiPT

Abbreviations / Short names of drugs shall be labelled. e.g. XTC, BTH or

This listing is for ^{Drug} Furanylethylfentanyl ^{Drug} FUEF . Crystal . Effects similar to ^{Drug} Heroin
 / ^{Drug} Fentanyl . ^{Drug} Furanylethylfentanyl is a is an ^{Drug} opioid RC and an analog of
^{Drug} fentanyl that is reported to be slightly weaker than ^{Drug} furanylfentanyl . It is also
 a small crystal instead of a powder and as such is less soluble in water . It
 is cheaper than ^{Drug} furanylfentanyl . Please check our vendor page for our policies
 regarding ordering , reships , etc . PRICING 1g ^{Drug} FUEF Crystal 35 - 35 / g

Context

1g Furanylethylfentanyl (FUEF) Crystal

Active Agents (The ingredient which makes you high) shall be labelled as drugs. e.g. CBD, THC, Diazepam or DMT

100 x. 05g cartridges filled with the highest quality distillate oil , 16 each - ORGANIC - Solvent FREE .

These are the most potent cartridges . Testing at approximately 90 ^{Drug}THC , these carts deliver the best vaping

experience on the market 500 ^{Drug}THC in 1 cartridge . Available flavors . - Strawberry Cheesecake - Guava

Official marketing names ARE drugs

This includes all kinds of marketing names in different countries or commercial marijuana strain names

^{Drug}ARTVIGIL - ^{Drug}armodafinil 150mg . Manufacturer HAB Pharmaceuticals . ^{Drug}Armodafinil is the more efficient successor

to ^{Drug}Modafinil . While you would take 200 mg of ^{Drug}Modafinil to achieve a certain level of increased mental

activity , you would only have to take 150 mg of ^{Drug}Armodafinil to achieve a comparable effect . People also

speak of ^{Drug}Armodafinil giving a stronger , sharper buzz , while ^{Drug}Modafinil feels softer , warmer , and slightly

Context

ARTVIGIL (generic NUVIGIL) armodafinil 150mg - 10

Drug Classes are Drugs

- Benzodiazepines, Opiates, Steroids, Indica / Sativa / Hybrid - ARE Drugs

^{Drug}Alprazolam is in a class of drugs called ^{Drug}benzodiazepines . It affects chemicals in

Slang names ARE drugs e.g. ganja or dope

Description Platinum and Top Shelf ^{Drug}Gorilla Glue 3 LB - Pure profit .

^{Drug}Gorilla . Glue is one of the most popular strains in the US right now . A

potent and high - yielding ^{Drug}hybrid , this bud produces a heavy yet .

comfortable high that knocks away pain . Contact Us at . KIK highlord 6

wickr highlord . What you see is what you get 100 Real Pics . Stealth

Shipping on every package marijuana , weed , kush , wax , oil ,

^{Drug}granddaddy purple , ^{Drug}fruity pebbles , ^{Drug}ganja , ^{Drug}cannabis , ^{Drug}OG kush , ^{Drug}afgahn kush

, mine , ^{Drug}weed , ^{Drug}dope , smoke , ^{Drug}cannabis , ^{Drug}ganja , high , ^{Drug}marijuana ,

^{Drug}mary jane , stoner , stoned , pot , herb , hemp , ^{Drug}hash , 420 , ^{Drug}kush , ^{Drug}haze

, dank , buds , spliff , bong , blunt , joint . All orders will go out the

following day . CONTACTS US . KIK highlord 6 wickr highlord .

Context

Platinum and Top Shelf Gorilla Glue 3 LB

Legal Drugs ARE Drugs

- In this example its alcohol, but it could also be Caffeine, Aspirin or tabacco.

LIQUID ^{Drug}GHB SYRUP THIS ^{Drug}GHB IS READY FOR USE - NO PREPPING REQUIRED
STRONG PURE . 99 ,95 PURE BASF ^{Drug}GHB . ULTIMATE STEALTH AAA .
DELIVERY GUARANTEED . NEVER COMBINE WITH ^{Drug}ALCOHOL CHECK FAQ .
BEFORE USE .

Context

GHB LIQUID SYRUP 250ml GHB (BASF)

df drug is mentioned but doesn't clearly indicate which drug it is, do NOT label it. (e.g. Orange Red bull Pills)

"Fishscale Powder", "These Mitsubishi Pills" or "This strain" will refer to a drug but its not clear, which drug it is exactly. In this case only XTC IS a drug and pills is NOT a drug.

Hello people if you want to try out the real dutch ^{Drug}xtc , we recommend to try
our orange redbulls . They are high quality pills with good presses and also
strong , so i recommend to take it in halves when you try for the first time .

Context

100X ORANGE REDBULLS XTC 280MG

The form / strength / adjectives of the drug shall NOT be labelled.

- At Ecstasy(XTC) it doesn't matter if they are Green Mitsubishi with 200mg

ARE YOU READY TO PARTY . BLUE ICE ^{Drug}EXCTASY PILLS GUARANTEED TO
MAKE YOUR NIGHT AN UNFORGETTABLE ONE . LETS GET THE PARTY

Context

@@\$ \$10 BLUE ICE EXCTASY PILL @@ FREE USA -USA 2

- Or that they have the imprint "OT-20" or "V-3605" on them. Those are NOT drugs.

The Producer or Manufacturer of the drug, is NOT a Drug

This also excludes creations from Drugdealers like Drcokes megasniff or MegaBelt IsolatorVapes. None of those are drugs.

In the example below Helix Pharma is only the producer and superdrol is the drug.

1 bottle of Helix Pharma ^{Drug}superdrol 10 mg x 100 units .

Context

HELIX PHARMA SUPERDROL 10MG X 100 UNITS

Natural chemicals which are part of your body are NOT a drug.

- Dopamin or estrogen do something to your body, but are NOT produced as drug and therefore we DON'T label them.

Processed Items like edibles are not always a drug.

Only the words which clearly refer to a drug are drugs. Brownies, Vapes or Candies are by itself not a drug, only the form!

NO DRUG in the TEXT?? - IMPORTANT

- If you cannot find any drug in the text, just annotate the first word with the tag "NONE" and continue.

None
This is for anyone who wants to share the love . This is also for situations
when a package is reshaped and both show up and you feel like paying for the
Context
Tip Jar Aka IOU Payments

HOW TO LABEL? - IMPORTANT

We want to label tokens which belong together as ONE single span:

Two tokens which are part of the same drug e.g. the "Master Yoda" Cannabis strain shall be labelled as single span. (Same goes for OG Kush or Master Kush etc.)

Drug Master Yoda Medical Grade Drug Indica , strong medicine Drug Master Yoda is . This strain
, however , is not about brute Force , as the original Drug Master Yoda would know
, and the Drug sativa side of this mostly Drug Indica buzz has just enough of an uplifting
balance to please just about anyone . Drug Master Yoda clinched 1st Place at High
Times Drug Cannabis Cup in 2012 , where the Drug OG Kush and Drug Master Kush cross was
already a local favorite . Growers also like this strain for its hybrid resilience
and its 8 week flowering time . As always our shipments are vacuum sealed
with zero smell .

Context
14g Master Yoda Medical Grade Indica

If there is a character (e.g. '!' or '-') inside of drug name -- include it in the labelling. The labels are not allowed to be separated!

Drug
Gorilla . Glue is one of the most popular strains in the US right now . A

Are you unsure about what is a drug? No problem - RE-OPEN this wholesome description for details or check the info of the "Drug" Class.



Test Question Review

When you get a test question wrong, you'll be able to review your mistakes in a special version of the tool. The errors you made will be underlined in red.

Some of your answers weren't what we expected. Please review the following messages so that you can get the next items correct!

Not Passing
0 of 3 tokens correct
(3 of 3 required)
A token is correct if it gets merged as required and gets a passing class label.

Incorrect Annotation
provided class:
none/other ('long')
correct classes:
none ('long')

Their **long** association with humans has led dogs to be uniquely attuned to human behavior and they are able to thrive on a starch - rich diet that would be inadequate for other canid species .

Context
The domestic dog (*Canis lupus familiaris* when considered a subspecies of the wolf or *Canis familiaris* when considered a distinct species) is a member of the genus *Canis* (canines), which forms part of the wolf-like canids, and is the most widely abundant terrestrial carnivore. The dog and the extant gray wolf are sister taxa as modern wolves are not closely related to the wolves that were first domesticated, which implies that the direct ancestor of the dog is extinct. The dog was the first species to be domesticated and has been selectively bred over millennia for various behaviors, sensory capabilities, and physical attributes. **Their long association with humans has led dogs to be uniquely attuned to human behavior and they are able to thrive on a starch-rich diet that would be inadequate for other canid species.** Dogs vary widely in shape, size and colors. They perform many roles for humans, such as hunting, herding, pulling loads, protection, assisting police and military, companionship and, more recently, aiding disabled people and therapeutic roles. This influence on human society has given them the sobriquet of "man's best friend".

Using this tool
Select a span (a word or group of words) by clicking on it. Once a span is selected, apply an annotation by clicking on one of the classes to the left.

Shortcuts
Some functions may not be available, depending on the design of this job.

Hold **SHIFT** + click (or click and drag across spans) to select multiple spans.

Double click a span to select it and all matching spans.

DELETE to remove annotations from selected spans.

B to break up selected spans into smaller spans.

M to merge selected adjacent spans into larger spans.

L to look up the selected span in search engine.

CMD/CTRL + Z to undo.

CMD/CTRL + Y to redo.

F to expand/contract fullscreen.

Annotator Guidelines Amazon Mechanical Turk

Drug Labelling

In this HIT we want you label all text spans, which clearly point to a specific drug or drug type.

HOW TO DO THIS?

1. **Read the Context** above the text to get an idea of which items could be contained in the text
2. **Read the provided drug description and label ALL spans which clearly identify a drug**
3. Are you unsure about what is a drug? We provided an extensive description, just press on "Instructions" and go to "more instructions".

Keep in mind: **Each span should Clearly identify a specific drug or drug type!**

Practical examples are:

- Clearly represent a specific drug e.g. hash, meth
- Chemical Compounds e.g. "3, 4 - methylphenidate"
- Short names e.g. XTC
- Agents e.g. THC, CBD or Diazepam
- Slang drug names: Coke, Fent
- Drug categories e.g. Indica/hybrid, benzodiazepines, opiates
- Pill imprints e.g. "V-3923"
- Chemical Details which extend the drug e.g. Viagra Sildenafil Citrate
- Legal or partial legal drugs e.g. "Alcohol", "Testosterone Enthalate", "Aspirin" or "Viagra"

DO NOT label tokens like:

- Description/Adjectives like drug strength, colour, form (powder/pill..) or looks etc.
- Form of Drug: e.g. powder, pill, paste
- Vague references like "this pill", "My Super Chocolate Bars" or "indoor bud"
- Producer of drugs e.g. Pfizer
- Natural Chemicals in your body e.g. estrogen/dopamin

Instructions

Shortcuts



101608 - 1 x SUBUTEX BUPRENORPHINE 8mg by INDIVIOR (ORIGINA

IN STOCK. This listing is for 1 pill 8 mg in original blister packing.

Name Subutex Manufacturer INDIVIOR. Active substance.

Buprenorphine Origin FRANCE. This listing is for 1 PILL 8 mg. If you are looking for a different amount, check my other listings in my store.

Labels



Drug

1

No entities to label

Submit

Note: The long version of the annotation guidelines was accessible over the „Instructions“ button. The long version was equal to the long Annotation guidelines from Appen.

List of Figures

3.1	An example for a drug item listing on a Darknet market platform called White House Market. Accessed on the 12.04.2021.	15
3.2	Crowd-sourcing project structure re-used from [SBDS14, p. 2]	16
3.3	Screenshot from the Appen labelling tool.	17
3.4	Lengths of drug item listings in final dataset	22
3.5	Word-cloud of the most occurring spans annotated as drugs, scaled by their occurrence.	22
3.6	Relative contribution of spans, clustered by the amount of their occurrence, to the total amount of annotations.	23
3.7	Model architecture used for our NER task	26
4.1	Screenshot of the Streamlit WebApp for comparing all predictions where False Negatives were found.	38
A.1	Agreement according to Cohen’s kappa from [VG05]	52

List of Tables

3.1	Inter-Annotator Agreement measures for the overall annotations by crowdworkers of Amazon MTurk and Appen.	20
3.2	Performance evaluation of the review process	20
3.3	Key facts of datasets	21
3.4	Descriptive Statistics of dataset	21
3.5	Results from splitting method evaluation	28
4.1	Model performance during hyper-parameter tuning using 100 rows for training in the Few-Shot scenario in the upper part and the full training set in the lower part of the table. Bold font marks the best model for each setting.	34
4.2	Full Training Set: Results after training on a different NER dataset in advance. Bold font marks the best and second best model. The second best model achieves a comparable performance, without any pre-training on a different dataset, and is therefore highlighted as well.	35
4.3	Few-Shot: Results after training on a different NER dataset in advance. Bold font marks the best model.	36
4.4	The first row presents the number of individual tokens per dataset. Row number two shows the vocabulary overlap between our corpus and the pre-training NER datasets used. Overlap is defined as the amount of unique words contained in both datasets divided by the total amount of unique words from both datasets. The last row contains the percentage of tokens marked as drug in our dataset contained in the pre-training NER datasets.	37
4.5	Final Results of the most important models in a Few-Shot setting evaluated on the test set	37
4.6	Final Results of the most important models using the full training dataset evaluated on the test set	38
A.1	Parameters and results of Appen crowdsourcing campaigns.	52
A.2	Performance of Crowdworkers IAA measured using Cohen' Kappa in [FSLR ⁺ 18]. The two values are reported for different annotation conditions to evaluate performance impact factors.	53
A.3	Parameters and results of MTurk crowdsourcing campaigns.	57