Chapter 1
Improving Adverse Drug Event Extraction with SpanBERT on Different Text Typologies

Beatrice Portelli*, Daniele Passabì*, Edoardo Lenzi*, Giuseppe Serra*, Enrico Santus[†] and Emmanuele Chersoni[‡]

* University of Udine, Udine (UD), Italy
[†] CSAIL MIT, Cambridge (MA), United States of America
[‡] The Hong Kong Polytechnic University, Hong Kong

portelli.beatrice@spes.uniud.it
passabi.daniele@spes.uniud.it
lenzi.edoardo@spes.uniud.it
giuseppe.serra@uniud.it
esantus@csail.mit.edu
emmanuele.chersoni@polyu.edu.hk

Abstract In recent years, Internet users are reporting Adverse Drug Events (ADE) on social media, blogs and health forums. Because of the large volume of reports, pharmacovigilance is seeking to resort to NLP to monitor these outlets. We propose for the first time the use of the SpanBERT architecture for the task of ADE extraction: this new version of the popular BERT transformer showed improved capabilities with multi-token text spans. We validate our hypothesis with experiments on two datasets (SMM4H and CADEC) with different text typologies (tweets and blog posts), finding that SpanBERT combined with a CRF outperforms all the competitors on both of them.

1.1 Introduction

Regulators, such as the Food and Drug Administration (FDA) and the European Medicine Agency (EMA), approve every year dozens of drugs, after verifying their safety in clinical trials. Sometimes, however, clinical trials are not sufficient to discover all potential Adverse Drug Events (ADE). Pharmacovigilance, therefore, monitors the drugs in the market to ensure that unexpected effects are timely identified and actions are taken to minimize their harm. A constantly growing number of patients chooses to describe the side effects on social media platforms, health forums and similar outlets. The growing amount of available ADE information in such sources and the high cost of manual extraction have recently made the automation of ADE extraction an important research topic in the natural language processing community (NLP), also encouraging a dedicated series of shared tasks [20, 24, 23]. Such extraction poses two major challenges, namely (i) the identification of relevant content among a large amount of unrelated posts; and (ii) the understanding of symptoms even when they are expressed in layman's terms.

In this paper we address the automation of ADE extraction through a tagging task: given a text, our goal is to identify whether it contains an ADE and its precise span. Because this task is often propaedeutic to other tasks (e.g. the mapping of the extracted ADEs to medical ontologies: see [23]) it is of key importance that the relevant span of tokens is identified as precisely as possible.

To this end, we apply for the first time SpanBERT [10], a recently proposed version of BERT [6] which has shown improved capabilities on multi-token text spans, such as those describing ADE. The performance of the system is evaluated on two widely-used benchmark datasets, containing two different kinds of texts: (i) short and highly informal tweets, in SMM4H [23]; and longer and more structured blog posts in CADEC [11]. In our experiments, we prove that this approach obtains competitive performance not only compared to common baselines but also compared to state-of-the-art transformer-based architectures. Moreover, our results show that, when SpanBERT is combined with CRF, it outperforms all the other approaches on both types of texts.

1.2 Related Work

The extraction of ADE mentions from social media texts is a task which started receiving growing attention in the last few years. This was prompted by the increasing number of users who discuss their drug-related experiences online, in platforms such as Twitter. Studies like [19, 16, 5] were among the first to propose machine learning approaches for this task, either using traditional feature engineering or word embeddings-based solutions.

The problem gained visibility with the introduction of the Social Media Mining for Health Applications (SMM4H) Workshop and the shared tasks co-located with it. Methods based on neural networks became a more common choice [25, 17]. During the latest instance of SMM4H, co-located with ACL 2019, transformers-based architectures such as BERT [6] and BioBERT [13] were the building blocks of the top performing systems [2, 14, 15]. However, these recent shared tasks have focused on a very particular type of social media texts: tweets, which are short, highly informal and noisy texts. This led the research community to pay less attention to longer and more structured social media texts, like forum posts, which are another interesting text typology for this kind of task.

The most recent progresses on non-tweet based benchmarks include thoughtful analyses of ADE extraction framed as a Named Entity Recognition task with discontinuous entity mentions (see [3] for a complete overview). Such studies focus on the fact that the ADE mentions might be long and include unconventional descriptions, and they typically evaluate their approaches on more structured text typologies (i.e. texts from medical forums) [22, 4], while the issue of their performance on short and informal texts has not been addressed, to the best of our knowledge.

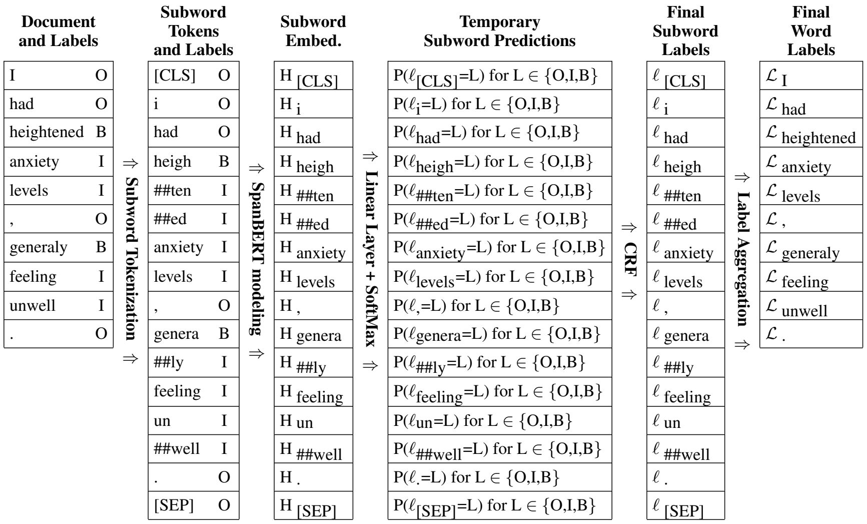

Fig. 1.1 Schema of the proposed SpanBERT+CRF architecture.

1.3 The Proposed Approach

Pre-trained methods, such as BERT [6] and BioBERT [13] have recently shown high gains in the ADE extraction task compared to previous approaches. However, these algorithms are trained in a self-supervised fashion by masking individual words or subwords units. In the ADE extraction task, longer spans and relations between multiple spans of text are instead the key. In addition, in short and highly contextual language, such as the one used in social media – which is characterized by acronyms, slang, metaphors, etc. – the meaning of each word in strongly influenced by the ones in its vicinity.

SpanBERT [10] takes into account wider portions of text by adopting both a different masking scheme and a different training objective. In fact, during training, random contiguous spans of tokens are masked, rather than individual words, forcing the model to predict the full span from the known tokens at its boundaries (the span-boundary objective - SBO). This basically means that the hidden representation of each word is encoding span-level information for its neighbours.

Since the SpanBERT model, to the best of our knowledge, has never been tested on the ADE extraction task [1], we propose a neural network architecture that combines SpanBERT with a Conditional Random Field classifier (henceforth CRF) [12, 18].

[1] Notice however that an architecture inspired by similar principles, SpERT, has been recently tested on ADE entity extraction and on the classification of drug-reaction relation, leading to improvements on both tasks [8].

Each sample goes through five main processing steps, which lead to the extraction of the ADE entities contained within the text. The following paragraphs describe these steps in detail, while Fig. 1.1 shows a schema of the architecture.

Preprocessing Each training sample is originally accompanied by a list of tuples which record the start- and the end-character of each ADE mention. As a first preprocessing step, this information is converted into the commonly used IOB (Inside-Begin-Outside) annotation schema. More precisely, each sentence is split into words and each word is labelled with one of the three letters: `B` if it is the beginning (first word) of an ADE mention; `I` if it is the continuation of an ADE mention; `O` if the word is outside of any ADE mention. For example, if we consider the sentence "I had heightened anxiety levels, generaly feeling unwell." (analyzed in Fig. 1.1) the phrase "heightened anxiety levels" is an ADE mention made of three words, which gets labeled as [`B`, `I`, `I`] respectively.

Subword Tokenization BERT-based models usually employ a tokenization method called Wordpiece tokenization [26], which allows to tokenize any word without incurring into "out-of-vocabulary" tokens. In particular, the vocabulary of the tokenizer is composed both of full words and wordpieces (sub-word units, which can be as short as single characters). Even if a word is not directly part of the vocabulary, it can always be split into a set of sub-words which belong to the vocabulary. For example the word "heightened" is not part of the SpanBERT vocabulary, but it can be split into three subwords: ["heigh", "##ten", "##ed"] (wordpieces can be identified by the leading ##). Since each full word was labelled using the IOB schema, we need to decide how to assign the labels to the eventual subword tokens. We choose to generate sub-labels which are consistent with the IOB schema: words labelled as `B` generate a series of subwords labelled as [`B`, `I`, . . . , `I`], while words labelled as `I` (or `O`) generate a series of identical `I` (or `O`) sub-labels.

In addition to the Wordpiece tokenization, BERT-base models usually employ additional special tokens to prepare the sentence before feeding it to the model: a `[CLS]` (classification) token that marks the beginning of the sample; a `[SEP]` (separator) token to mark the end of the sample; a `[PAD]` (padding) token to pad all input sequences to the same length (not shown in the image). All predictions generated on the `[PAD]` tokens are discarded and are not taken into consideration when computing the loss for the model.

SpanBERT modeling The first actual component of the architecture if a SpanBERT neural network, composed of 12 encoder layers with multi-head attention. The network takes as input the sequence of subword tokens $[x_i]$ and outputs a sequence of embeddings $[\mathrm{H}_{x_i}]$ (one for each subword token). Internally, each one of the transformer (encoder) layers computes an intermediate embedding of each token. The attention mechanism allows the network to enrich the embeddings with contextual information after each layer.

Intermediate label prediction The embedding of each token is processed by a linear layer, which projects the embedding of each subword token to a vector of size

3 (representing the IOB labels for the tokens). The softmax operator turns the raw value into a probability distribution over the possible output labels.

Conditional Random Field The CRF module takes as input the probability distribution generated by the SpanBERT model, and produces another sequence of subword-level IOB labels. This step aims at denoising the output labels produced by the previous components exploiting a well known statistical model, which was widely used for this task prior to the introduction of deep neural networks.

Label aggregation Finally, the subword-level predictions need to be aggregated to produce word-level labels which can be easily mapped to the original input sequence.

A set of subword labels $\{\ell_i\}$, is aggregated into a word label $\mathcal{L}$ using the first rule that applies: (i) if $\ell_i = \mathtt{O}$ for all i, then $\mathcal{L} = \mathtt{O}$; (ii) if $\ell_i = \mathtt{B}$ for any i, then $\mathcal{L} = \mathtt{B}$; (iii) if $\ell_i = \mathtt{I}$ for any i, then $\mathcal{L} = \mathtt{I}$. The aggregated output is therefore a sequence of word-level IOB labels, which can be compared to preprocessed inputs or easily converted into character level indices.

1.4 Experimental Results

1.4.1 Datasets

The two datasets chosen for the experiments two widely used ADE extraction datasets. SMM4H is the training dataset for Task 2 of the Shared Task SMM4H 2019 [23]. It contains 2276 tweets which mention at least one drug name (1300 tweets are positive for the presence of ADEs while the other 976 are negative control samples). We further partitioned the training set into training and validation sets (80:20), maintaining the proportions of positive and negative samples. The competition also includes a blind test set composed of 1573 unlabelled tweets. The final metrics are evaluated on the blind test set using the scoring system provided by the competition[2]. This ensures that the results are comparable with all the systems which took part in the shared task. CADEC [11] is a widely used and publicly available ADE extraction dataset. It contains 1250 posts from the health-related forum "Aska-Patient", annotated for the presence of ADEs. We use the same training, validation and test splits made publicly available by [4].

1.4.2 Metrics

As evaluation metrics we use the Strict and Partial (also known as Relaxed) F1-Score, which are commonly adopted for this task [21]. They are computed on entity level (as opposed to the token level). The partial score takes into account "partial"

[2] https://competitions.codalab.org/competitions/20798

matches, in which it is sufficient for a prediction to partially overlap with the gold annotation. On the other hand, the Strict score requires a perfect match between the system output and the annotated ADE.

1.4.3 Baselines

We directly compare our proposed SpanBERT+CRF on the two datasets with four different architectures.

BERT (version base, uncased)[3], a Transformers-based baseline finetuned on the training data. The outputs of the final hidden layer are passed to a linear layer which projects the hidden representation to the needed output space (one label per token).

BERT+CRF the same BERT model but with a CRF classifier on top, used to denoise the output of BERT as we described in Section 1.3.

SpanBERT (version base, cased) is the simpler version of the proposed model, with only a linear layer on top.

TMRLeiden [7]: 4th classified in the ADE-extraction task at the SMM4H 2019 Workshop [23]. It is composed of a BiLSTM network taking as input a concatenation of BERT embeddings and Flair embeddings [1]. It was chosen for direct comparison since the first four systems in the competition achieved comparable results (TMR-Leiden had the second best Strict F1-Score) and TMRLeiden was the only top-team which made their code public, which allows us to compare their system with others (and on different datasets) with only minimal adjustments.

We perform a partial comparison with the top-3 classified at SMM4H 2019 (*KFU NLP* [15], THU_NGN [9], MIDAS@IHTD [14]) by reporting their scores on the SMM4H dataset as published in the overview of the workshop [23].

1.4.4 Implementation Details

TMRLeiden was reimplemented using its original code as a base[4] and trained according to the details in its paper.

For the transformer-based architectures we performed a parameter-tuning analysis via grid-search using different learning rates ($[5e{-}4, 5e{-}5, 5e{-}6]$) and dropout rates (from 0.15 to 0.30, increments of 0.05). All the architectures were trained for 50 epochs on the training set. The best learning rate, dropout rate and maximum epoch were chosen evaluating the models on the validation set and using the Partial F1-score as the performance metric. The models were trained using the best hyperparameters on the concatenation of the training set and the validation set. This procedure was repeated five times with different random seeds and the results on the test set have been averaged.

[3] Preliminary experiments with cased and uncased versions showed that the latter performed better.

[4] https://github.com/AnneDirkson/SharedTaskSMM4H2019

Table 1.1 Strict and Partial F1-Score with standard deviations for all architectures on both datasets

Architecture	Partial F1 CADEC	Partial F1 SMM4H	Strict F1 CADEC	Strict F1 SMM4H
BERT	77.7 ± 0.3	68.7 ± 0.4	65.2 ± 0.5	43.5 ± 0.4
BERT+CRF	77.2 ± 0.5	**69.8** ± **0.2**	64.4 ± 0.8	45.8 ± 0.9
SpanBERT	79.2 ± 0.6	65.8 ± 2.2	67.2 ± 0.8	47.1 ± 1.9
SpanBERT+CRF	**79.4** ± **0.3**	67.7 ± 0.7	**67.6** ± **0.6**	**47.5** ± **0.2**
TMRLeiden [7]	77.1 ± 0.8	62.5	65.0 ± 1.1	43.1
KFU NLP [15]	–	65.8	–	46.4
THU_NGN [9]	–	65.3	–	35.6
MIDAS@IIITD [14]	–	64.1	–	32.8

1.4.5 Evaluation

Table 1.1 presents a comparative evaluation between our architecture and the other approaches on both datasets (we reported the scores of the shared task 2019 for the systems whose code is not publicly available). Results are partitioned into two sections, one for each metric.

Looking at the results on CADEC, we can see that SpanBERT+CRF achieves the best performance for both metrics, showing that our approach is able to properly deal with the long texts contained in forum posts. SpanBERT (without CRF) gets the second place, while TMRLeiden achieves a performance slightly lower than BERT, and with an higher standard deviation. This could be explained by the fact that TMRLeiden was specifically designed to perform this task on tweets, and does not express its full potential on forum posts. This finding confirms that it is worth testing ADE systems on different textual typologies, since text genre has an important impact on the performance.

Considering the results on the SMM4H dataset: we see that the proposed SpanBERT+CRF achieves the best performance on the Strict metric, while BERT-based architectures perform better on the Partial metric. The BERT-based variants only surpass SpanBERT+CRF in this particular case, and only on the tweets data. Interestingly, all these simple architectures improve over the performances of all the top-4 systems that took part in the SMM4H 2019 shared task.

More in general, models tend to perform better on CADEC. This is most likely due to the larger size of the dataset and to the more regular type of text, compared to the noisy tweets. We can also notice that the denoising action of the CRF seems to be effective in most situations (e.g. the scores for the simple SpanBERT are lower than the ones achieved by SpanBERT+CRF).

We take this as a preliminary evidence that SpanBERT is a particularly convenient solution for dealing with ADE, especially for longer and more structured texts, and that it is more precise in detecting the ADE boundaries, as suggested by the more marked improvements for the Strict metric.

Table 1.2 p-values for the McNemar test on all pairs of architectures on the two datasets

Architecture 1	Architecture 2	SMM4H	CADEC
BERT	BERT+CRF	**< 0.001**	**< 0.001**
BERT	SpanBERT	**< 0.001**	≥ 0.1
BERT	SpanBERT+CRF	**< 0.001**	≥ 0.1
BERT	TMRLeiden	**< 0.001**	**< 0.001**
BERT+CRF	SpanBERT	**< 0.001**	**< 0.001**
BERT+CRF	SpanBERT+CRF	≥ 0.1	≥ 0.1
BERT+CRF	TMRLeiden	**< 0.001**	**< 0.001**
SpanBERT	SpanBERT+CRF	**< 0.001**	≥ 0.1
SpanBERT	TMRLeiden	**< 0.01**	**< 0.001**
SpanBERT+CRF	TMRLeiden	**< 0.001**	**< 0.001**

1.4.6 Quantitative Analysis

We performed a series on quantitative tests comparing the sets of ADE entities predicted by the various models. The objective was to investigate whether the different architectures produce significantly different results.

True Positive and False Positive Distribution First of all we compared the true positive and false positive predictions for all the models using the McNemar test. Since the gold labels for the SMM4H test set are not public, we compared the predictions generated on the validation set (training all the models on the training set only).

The p-values obtained by comparing each pair of models are summarized in Table 1.2. The differences are statistically significant (in most of the cases with $p < 0.001$) on the SMM4H dataset, with the exception of the comparison between SpanBERT+CRF and BERT+CRF. Indeed, even if the two models are different in terms of overall strict F1 score (as shown in Table 1.1), they are really close in terms of precision and mainly differ due to BERT+CRF having a lower recall. The situation is less defined on the CADEC dataset, with the differences being clearly significant only when comparing BERT-based-models with the baseline TMRLeiden. This is due to all these models reporting similar precision values on CADEC (while they have different recall values, which determined the differences in the final F1 scores in Table 1.1).

Text Statistics of the Predictions We also computed a series of text-specific metrics on all the predictions generated by the models on the test data.

The chosen metrics are: three widely used readability indices (Dale–Chall readability score, Automated readability index, Flesch–Kincaid reading ease) and two measures of text length (Syllable Count and Character Length). The readability indices indicate whether the extracted text contains words which are common or easy to read in the English language. In this case the presence of uncommon or difficult-to-read words could be indicative of the use of medical terms or uncommon expressions in the predicted entities. Measuring the length of the whole predictions

Table 1.3 Average text metrics for the BERT-based models on the SMM4H dataset

	BERT	BERT+CRF	SpanBERT	SpanBERT+CRF
Dale Chall Readability^{+}	**08.34 ± 0.24**	08.15 ± 0.25	08.20 ± 0.22	08.32 ± 0.24
Automated Readability^{+}	08.42 ± 0.31	08.37 ± 0.31	08.35 ± 0.28	**08.51 ± 0.30**
Flesch Reading Ease^{-}	62.73 ± 1.98	63.57 ± 2.08	**62.60 ± 1.82**	62.92 ± 1.97
Syllable Count^{+}	02.49 ± 0.04	02.46 ± 0.04	**02.89 ± 0.05**	02.79 ± 0.05
Character Length^{+}	09.66 ± 0.14	09.60 ± 0.15	**11.29 ± 0.18**	10.94 ± 0.18

$^{+}$ Higher means less common $^{-}$ Lower means less common

by syllables or by characters is also an indicator of the complexity of the terms and the ability of the model to predict significant contiguous spans of text.

Table 1.3 summarizes the metrics for all the architectures. We can see that the predictions generated by the SpanBERT-based models show an higher frequency of uncommon (or difficult-to-read) words for all the metrics: lower readability scores and longer predictions. To test whether these differences were significant we used the non parametric Mann-Whitney-U significance test on each pair of models. The test shows that:

1. when comparing two architectures based on the same model (e.g. BERT with and without CRF) differences are not significant, for all the considered metrics. This means that the addition of the CRF enhances the performances of the model without altering the textual style of the predictions.
2. when comparing BERT-based architectures with SpanBERT-based architectures Syllable Count and Character Length are significantly different ($p < 0.001$). This means that using a model which was pretrained with span-level knowledge affects the length of the final predictions.
3. when comparing BERT-based architectures with SpanBERT-based architectures Dale Chall Readability and Flesch Reading Ease are significantly different ($p < 0.05$). In particular SpanBERT+CRF predicts entities containing less common words compared with the predictions of BERT+CRF.

Both analyses lead to the conclusion that the predictions of the architectures are significantly different, and that SpanBERT-based models predict longer entity mentions containing uncommon or "difficult" words.

1.4.7 Qualitative Error Analysis

To illustrate the differences between our SpanBERT+CRF solution and a tweets-focused architecture such as TMRLeiden, we propose a qualitative error analysis (see Table 1.4). We selected the samples on which one of the architectures performed clearly better than the other one in terms of Strict F1-Score, and analyzed the results manually. In particular, we observe that our solution has the tendency to extract longer ADE spans compared to TMRLeiden, and it is better at dividing consecutive ADE

Table 1.4 Examples of errors: gold annotations bold and enclosed in squared brackets, entities extracted by SpanBERT+CRF overlined, by TMRLeiden underlined

#	Text
1	[**short term memory affected**], [**loss of concentration levels**], [**dizzy**], [**hightened anxiety levels**], [**generaly feeling unwell like walking in a daze (zombie like)**] [**weakness in leg muscles**].
2	this lozenge is making my whole mouth [**numb**]... is that normal?
3	user fluoxetine and quet combo [**zombified**] me... ah, the meds merrygoround bipolar
4	it feels like my [**brain is melting**]. numbermg of vyvanse in number hours... not smart danielle. not smart.
5	i have a slipped vertabrae and a degenerative disk. this medicine has been extrememly helpful.
6	i never had bleeding or vomiting just alot alot of [**fatigue**] and face was [**pale**] and [**lost tons of weight**].

spans (Table 1.4, Example 1). This leads to better results on the CADEC datasets, which contains longer descriptions, but a worse precision on SMM4H, where the annotations are often one-word long (Example 2). However, sometimes this is more a matter of annotation choice ("whole mouth numb" in Example 2 is actually a more precise description of the ADE) and it impacts only the Strict evaluation metric. SpanBERT+CRF seems also able to identify highly informal/slang descriptions ("zombified", "jacked up", "brain melting", etc.), whereas TMRLeiden completely ignores them (Examples 3,4).

As regards false positives, both models make similar errors, misclassifying as ADE some mentions of diseases (Example 5), symptoms generated by the diseases themselves and repercussions of the ADE (e.g. extracting both "joint pain in shoulder", the ADE, and "could not lift my arms over my head", which is just a consequence of it). This problem might be addressed by adding more output classes to the models, training them to jointly detect ADEs, Diseases, and Symptoms that are not connected to Drugs.

For both models, further errors are generated by the lack of proper handling of negation, and therefore samples reporting the absence of a specific ADE are erroneously treated as mentions of such ADE (Example 6).

1.5 Conclusions

We introduced the SpanBERT architecture to the task of ADE extraction from different types of texts. We carried out our experiments on two widely-used datasets, respectively containing informal tweets and longer blog posts, SpanBERT obtained competitive results for both Strict and Partial F1-score with state-of-the-art models, outperforming all of them in combination with a CRF for the Strict metric. The error analysis also suggests that SpanBERT is more capable of modeling the presence of uncommon words and longer entity mentions when compared to the other approaches. Our results show the flexibility of the model on different typologies of data, and its higher accuracy in modeling the span boundaries of the ADEs.

References

1. Akbik, A., Bergmann, T., Blythe, D., Rasul, K., Schweter, S., Vollgraf, R.: FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP. In: Proc. of NAACL 2019 (Demonstrations), pp. 54–59. Association for Computational Linguistics, Minneapolis, Minnesota (2019)
2. Chen, S., Huang, Y., Huang, X., Qin, H., Yan, J., Tang, B.: HITSZ-ICRC: A Report for SMM4H Shared Task 2019-Automatic Classification and Extraction of Adverse Effect Mentions in Tweets. In: Proc. of the Fourth SMM4H Workshop & Shared Task, pp. 47–51. Association for Computational Linguistics, Florence, Italy (2019)
3. Dai, X.: Recognizing Complex Entity Mentions: A Review and Future Directions. In: Proc. of ACL 2018, Student Research Workshop, pp. 37–44. Association for Computational Linguistics, Melbourne, Australia (2018)
4. Dai, X., Karimi, S., Hachey, B., Paris, C.: An Effective Transition-based Model for Discontinuous NER. In: Proc. of the 58th Annual Meeting of the ACL, pp. 5860–5870. Association for Computational Linguistics, Online (2020)
5. Daniulaityte, R., Chen, L., Lamy, F., Carlson, R., Thirunarayan, K., Sheth, A.: "When 'Bad' Is 'Good'": Identifying Personal Communication and Sentiment in Drug-related Tweets. JMIR Public Health Surveill. **2**(2), e162 (2016)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proc. of NAACL-HLT 2019, vol. 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (2019)
7. Dirkson, A., Verberne, S.: Transfer Learning for Health-related Twitter Data. In: Proc. of the Fourth SMM4H Workshop & Shared Task, pp. 89–92. Association for Computational Linguistics, Florence, Italy (2019)
8. Eberts, M., Ulges, A.: Span-based Joint Entity and Relation Extraction with Transformer Pre-training. In: Proc. of ECAI 2020, *Frontiers in Artificial Intelligence and Applications*, vol. 325, pp. 2006–2013. IOS Press (2020)
9. Ge, S., Qi, T., Wu, C., Huang, Y.: Detecting and Extracting of Adverse Drug Reaction Mentioning Tweets with Multi-Head Self Attention. In: Proc. of the Fourth SMM4H Workshop & Shared Task, pp. 96–98. Association for Computational Linguistics, Florence, Italy (2019)
10. Joshi, M., Chen, D., Liu, Y., Weld, D.S., Zettlemoyer, L., Levy, O.: SpanBERT: Improving Pre-training by Representing and Predicting Spans. Transactions of the Association for Computational Linguistics **8**, 64–77 (2020)
11. Karimi, S., Metke-Jimenez, A., Kemp, M., Wang, C.: Cadec: A Corpus of Adverse Drug Event Annotations. J. Biomed. Inform. **55**, 73–81 (2015)
12. Lafferty, J., Mccallum, A., Pereira, F.: Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In: Proc. of ICML 2001, pp. 282—-289. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2001)
13. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C.H., Kang, J.: BioBERT: A Pre-trained Biomedical Language Representation Model for Biomedical Text Mining. Bioinformatics **36**(4), 1234–1240 (2020)

14. Mahata, D., Anand, S., Zhang, H., Shahid, S., Mehnaz, L., Kumar, Y., Shah, R.R.: MIDAS@SMM4H-2019: Identifying Adverse Drug Reactions and Personal Health Experience Mentions from Twitter. In: Proc. of the Fourth SMM4H Workshop & Shared Task, pp. 127–132. Association for Computational Linguistics, Florence, Italy (2019)
15. Miftahutdinov, Z., Alimova, I., Tutubalina, E.: KFU NLP Team at SMM4H 2019 Tasks: Want to Extract Adverse Drugs Reactions from Tweets? BERT to The Rescue. In: Proc. of the Fourth SMM4H Workshop & Shared Task, pp. 52–57. Association for Computational Linguistics, Florence, Italy (2019)
16. Nikfarjam, A., Sarker, A., O'Connor, K., Ginn, R.E., Gonzalez-Hernandez, G.: Pharmacovigilance from Social Media: Mining Adverse Drug Reaction Mentions Using Sequence Labeling with Word Embedding Cluster Features. J. Am. Med. Inform. Assoc. **22**, 671–681 (2015)
17. Nikhil, N., Mundra, S.: Neural DrugNet. In: Proc. of the 2018 EMNLP Workshop SMM4H, pp. 48–49. Association for Computational Linguistics, Brussels, Belgium (2018)
18. Papay, S., Klinger, R., Padó, S.: Dissecting Span Identification Tasks with Performance Prediction. In: Proc. of EMNLP 2020, pp. 4881–4895. Association for Computational Linguistics, Online (2020)
19. Sarker, A., Gonzalez, G.: Portable Automatic Text Classification for Adverse Drug Reaction Detection via Multi-corpus Training. J. Biomed. Inform. **53**, 196–207 (2015)
20. Sarker, A., Gonzalez-Hernandez, G.: Overview of the Second Social Media Mining for Health (SMM4H) Shared Tasks at AMIA 2017. In: Proc. of SMM4H@AMIA 2017, vol. 1996, pp. 43–48. CEUR Workshop Proceedings (2017)
21. Segura-Bedmar, I., Martínez, P., Herrero-Zazo, M.: SemEval-2013 Task 9 : Extraction of Drug-Drug Interactions from Biomedical Texts (DDIExtraction 2013). In: Proc. of SemEval 2013, pp. 341–350. Association for Computational Linguistics, Atlanta, Georgia, USA (2013)
22. Stanovsky, G., Gruhl, D., Mendes, P.: Recognizing Mentions of Adverse Drug Reaction in Social Media Using Knowledge-Infused Recurrent Models. In: Proc. of EACL 2017, vol. 1, Long Papers, pp. 142–151. Association for Computational Linguistics, Valencia, Spain (2017)
23. Weissenbacher, D., Sarker, A., Magge, A., Daughton, A., O'Connor, K., Paul, M., Gonzalez, G.: Overview of the Social Media Mining for Health (SMM4H) Shared Tasks at ACL 2019. In: Proc. of the Fourth SMM4H Workshop & Shared Task, pp. 21–30. Association for Computational Linguistics, Florence, Italy (2019)
24. Weissenbacher, D., Sarker, A., Paul, M., Gonzalez, G.: Overview of the Social Media Mining for Health (SMM4H) Shared Tasks at EMNLP 2018. In: Proc. of the 2018 EMNLP Workshop SMM4H, pp. 13–16. Association for Computational Linguistics, Brussels, Belgium (2018)
25. Wu, C., Wu, F., Liu, J., Wu, S., Huang, Y., Xie, X.: Detecting Tweets Mentioning Drug Name and Adverse Drug Reaction with Hierarchical Tweet Representation and Multi-Head Self-Attention. In: Proc. of the 2018 EMNLP Workshop SMM4H, pp. 34–37. Association for Computational Linguistics, Brussels, Belgium (2018)
26. Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Łukasz Kaiser, Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., Dean, J.: Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. CoRR **abs/1609.08144** (2016). URL http://arxiv.org/abs/1609.08144