Novel Transformer-based approaches for Adverse Drug Reaction identification in patient reviews: a comparative analysis

Karina Gabor, Marta Emili Garcia Segura, Teona Ristova, Giomaria Murgia

Abstract

Pharmacovigilance plays a critical role in ensuring drug safety through the detection, assessment, comprehension, and prevention of Adverse Drug Reactions (ADRs). Data mining and, more recently, Natural Language Processing and Machine Learning techniques have been the object of research in this domain. This study proposes two novel approaches for ADR extraction from patient drug reviews. The first suggested framework aims to take advantage of unannotated data to construct richer text representations. It does so by computing and clustering word embeddings, which are then used as inputs for a CRF to perform sequence labelling. The second suggested framework involves fine-tuning a BERT model for token classification. By leveraging the powerful capabilities of transformers architectures, significant improvements are demonstrated relative to previous studies and baseline data mining techniques. The CRF model also shows considerable progress when equipped with state-of-the-art contextual word embeddings. The findings highlight the potential of exploiting similar methodologies to enhance ADR identification by pharmaceutical companies and regulatory entities.

1 Introduction

In the pharmaceutical industry, significant research efforts are devoted to ensuring drugs undergoing the approval process are safe and effective for everyone to use. In the United States, the Food and Drug Administration (FDA) ensures drug safety by conducting clinical trials where adverse drug events (ADE) can be detected. An ADE could be an injury or unintended effect of a drug that has been recommended to a patient. Adverse drug reactions (ADRs) are a subset of ADEs that refer to unexpected side effect resulting from the regular use of a medication (Karimi et al., 2015b). ADRs have been identified as a significant public health concern due to their association with high mortality rates (Nikfarjam et al., 2015).

The FDA, or its respective drug regulatory body, monitors ADRs through clinical trials. These trials, however, typically involve a limited number of participants, are of short duration, and are not fully representative of the target population (Gopalakrishnan and Ramaswamy, 2017; Zeng et al., 2002; Stephens et al., 1985). For example, the first stage of the clinical trials conducted by the FDA includes 20-80 participants, while the final stage includes around 3000 participants (Meadows, 2002).

Given the present widespread internet access, numerous users share their opinions on drugs through online reviews (Dinh and Chakraborty, 2020). The aim of this project is to leverage this information to identify ADRs, which could offer a more efficient approach to detecting ADRs in conjunction with clinical trials.

Various techniques in information retrieval and data mining have already been utilised to detect ADRs (Wu et al., 2012; Cameron et al., 2013), however, this paper aims to investigate more contemporary natural language processing techniques. For instance, Nikfarjam et al. (2015) introduce a novel approach to extract ADRs using ADRMine, a concept extraction system based on machine learning that employs conditional random fields (CRFs). The paper's primary strength lies in the use of unlabelled data to construct richer representations of the scarce annotated data. The approach is particularly meaningful due to the vast amount of available unlabelled reviews and the costly and resource-intensive annotation process.

In the field of medicine, particularly in the domain of drug reviews, transformer models have demonstrated state-of-the-art performance in sentiment analysis tasks (Colón-Ruiz and Segura-Bedmar, 2020). Moreover, BERT (Bidirectional Encoder Representations from Transformers) models have contributed towards better results in the domain of sequence labelling (Tsai et al., 2019). The superior power of transformer models has motivated the direction of the paper to use BERT for ADR detection.

This paper proposes 2 new models for extracting ADRs. The first model is an extension of the work presented by Nikfarjam et al. (2015), where words are encoded using BERT in a feature-based approach, rather than using Word2Vec. The second model proposed is a BERT model for token classification, where the extraction of ADRs is approached as a sequence labelling task.

This paper is organized to first describe the related work on information retrieval and machine learning techniques used to identify ADRs in Section 2, followed by Section 3 which focuses on the methods used which includes a brief explanation of the models, and the data used. In Section 4, the experiments and the model setups are explained. Section 5 focuses on the results achieved by the models and compares their performance. Finally, Section 6 presents a conclusion and discusses the limitations and potential future work.

2 Related Work

Analysing unstructured data, such as online reviews, can be challenging due to the use of colloquial language, grammatical errors and misspellings.(Nikfarjam et al., 2015) Nevertheless, online reviews provide an opportunity to access first-hand, unfiltered, patient reports. For this reason, it has been an active field of research for over ten years.

Among the first examples of ADR extraction from patient health forums Leaman et al. (2010) collect user comments from healthcare forum DailyStrength.com aiming to identify adverse effects for six selected drugs. Their approach is based on a lexicon of adverse reactions collected from medical resources (e.g. Unified Medical Language System (Bodenreider (2004))) and augmented with a set of colloquial phrases. Terms from the lexicon were found in the user comments by comparing a sliding window of tokens from the comment to each token in the lexical term. With a different approach, Liu et al. (2011) deal with colloquial language in patient reviews by constructing a dictionary of common sentences from the reviews themselves. They then employ statistical techniques to compare word frequency distributions and uncover salient phrase patterns in the data. More recently, Liu and Chen (2013) designed the AZDrugMiner system, which uses statistical learning through MetaMap (a software which identifies meaningful biomedical concepts within a text, used mainly in the medical field) to detect ADEs in patient forum posts. They report an F-score of 66.9% for the ADE extraction task.

While results have been promising, researchers underline the need for progressively advanced computing algorithms to deal with the increasing data availability, as the current statistical and data-mining techniques are not up to the task and often require additional manual assessment, unsuitable to large-scale data (Karimi et al. (2015b)). To bridge this gap, newer Natural Language Processing and machine learning techniques have emerged as the standard for conducting research in this field. Bian et al. (2012) use NLP to analyse Twitter messages and extract relevant textual and semantic features based on concepts returned by the UMLS Metathesaurus (a large biomedical thesaurus that is organized by concept, or meaning, and links similar names for the same concept from different vocabularies (Bodenreider (2004))). They then employ Support Vector Machines to mine ADRs, obtaining limited performance.

In Nikfarjam et al. (2015) the authors introduce ADRMine, a machine learning-based sequence tagger for automatic extraction of ADR mentions from social media posts. A lexicon-based approach akin to those in the aforementioned paper is implemented as baseline and found to be suboptimal. The proposed method consists of constructing word embeddings with Word2Vec from a dataset of labelled and unlabelled reviews, which are then used to compute K-means clusterings. Cluster associations are subsequently employed as features in a Conditional Random Field model, which achieves

improved results with F-scores of 0.72 and 0.82 on two different corpora.

Another approach to unsupervised data can be found in Yates et al. (2015) where the difficulty of annotating social media on a large scale is addressed through an alternative evaluation scheme that takes advantage of the ADRs listed on drug labels. A combination of Multinomial Naive Bayes and CRF is then used on a much larger partially annotated corpus of social media posts. Their results show using known ADRs as pseudo-annotations can be beneficial in increasing precision but negatively impacts recall and F1 scores.

In recent years, the Transformer architecture introduced by Vaswani et al. (2017) has achieved remarkable performance on various natural language processing (NLP) tasks, such as machine translation, question-answering and language modelling. Afterwards, Devlin et al. (2018) presented BERT (Bidirectional Encoder Representations from Transformers), which surpassed previous benchmarks on eleven NLP tasks.

Transformer-based models and BERT modifications have shown significant improvements in sequence labelling tasks such as NER (Lee et al. (2020)). Hence, one of the aims of this paper is to achieve a comparatively higher performance relative to previously applied techniques in the context of ADR identification in online reviews by using a fine-tuned BERT model.

3 Methods

3.1 Data Processing

This project uses five distinct datasets. The first dataset comprises a medical term lexicon that is used to extract ADRs from the reviews using information retrieval models. The second dataset includes medical reviews that are unannotated and used for unsupervised K-Means clustering. The remaining three datasets consist of annotated medical reviews, which are used for supervised models, namely the fine-tuned BERT and CRF models. The decision to merge the datasets with annotated medical reviews was motivated by the absence of a single comprehensive annotated dataset. The aforementioned datasets are described below:

1. Lexicon with medical terms - contains 13,699 ADRs, as well as their unique identifiers and sources (Nikfarjam et al., 2015).

2. UCI ML Drug Review Dataset - comprises over 200,000 unannotated patient drug reviews (Gräßer et al., 2018).

3. CSIRO Adverse Drug Event Corpus (CADEC) - an annotated corpus of medical forum posts on patient-reported ADEs. It contains 1250 patient posts (Karimi et al., 2015a).

4. Psychiatric Treatment Adverse Reactions (PsyTAR) dataset - contains 891 patient reviews on the effectiveness and ADEs associated with psychiatric medications (Zolnoori et al., 2019).

5. Annotated Drug Reviews (ADR) Dataset - includes 247 annotated reviews by patients on adverse side effects.

Combination and standardization of the annotated datasets

The CADEC, PsyTAR, and ADR datasets all come from different sources. As such, the structure of the annotations in each of these datasets is different. The CADEC dataset labels each ADR by specifying its exact location within the review. For example, if a review mentions ”headache”, the corresponding label will indicate the location of an ADR to be between lines 18 and 25. By contrast, the PsyTAR dataset directly annotates each ADR in a review with its corresponding text.

The first step of the process is combining these three different datasets into a single standard dataset format. In the process, two datasets are formed.

The first dataset contains all the reviews, i.e. each row in this dataset is a single review. It has three columns, namely, *text id* (a unique review identifier), *text* (the full review in text format), and *dataset* (a column specifying the source of the dataset). The combined dataset contains 2388 rows, which is slightly fewer than the sum of the lengths of the three original datasets. This discrepancy is due to a few CADEC annotations being missing from the original files.

The second dataset is structured such that it contains a row for each of the ADRs. The columns included are *ADR* (the text form of the ADR), *txt_id* (the id of the review), *dataframe* (the source of the dataset), *start* (the start position of the ADR within the review), and *end* (the end position of the ADR within the review). The first and second datasets can be merged into a single one by using the *txt_id* column.

Tokenization and labelling

As outlined previously, the combined dataset provides the location of the ADRs within each review, the text of the review, and the text of the ADRs. To perform sequence labelling, the first step is to tokenize the text of the reviews and the ADRs. Two distinct tokenizations were used throughout this project. For the information retrieval techniques and the reproduction of the ADR-Mine model, the text of each review was lemmatized using the `nltk` library. Additionally, all the text was converted to lowercase and any non-alphabetic characters were removed. For the ADR-Mine extension and the BERT model, the text was tokenized using the BERT Tokenizer.

Once the tokens were extracted, a new *labels* column was generated by assigning a binary label to each token. The ADRs were labelled as 1 and the remaining tokens as 0.

3.2 Lexicon-Based Information Retrieval Models

Information Retrieval refers to the process of extracting relevant information from a large collection of data based on an input query. The collection of data is usually a corpus of documents. In this task, the drug reviews from the combined dataset serve as the queries and the list of symptoms (or ADRs) from the ADR Lexicon represents the document collection.

To establish a performance benchmark for the two proposed models, two widely used Information Retrieval algorithms - TF-IDF and BM25 - are employed to perform ADR extraction. These techniques aim to rank ADRs based on their relevance to each specific review.

TF-IDF

TF-IDF or Term Frequency-Inverse Document Frequency quantifies the importance of a word in a given document. It assign scores to each document in the corpus and retrieves the most relevant ones to the input query. By construction, the metric rewards occurrences of search words via the TF score and penalises the appearance of common words through the IDF score.

BM25

OkapiBM25 (Best Matching 25) is an improved TF-IDF method that computes the relevance score based on the probability that the user will consider the result relevant. The BM25 relevance score is influenced by several key factors. Firstly, the score is proportional to the TF- and IDF- scores, meaning that the frequency of the search words is rewarded, whilst the occurrence of common words is penalised. Secondly, the score is inversely proportional to the document length.

3.3 ADR-Mine and extended methods

ADR-Mine is a concept extraction model that uses Conditional Random Fields (CRFs). The effectiveness of the model is largely attributed to the choice of input features, which are generated using unsupervised learning methods. The model comprises three main components: encoding, clustering and predicting the labels of the tokens. The ADR-Mine model was replicated and an alternative encoding method that uses the BERT model in a feature-based approach was explored.

Constructing word embeddings

A word embedding is the representation of words in a high dimensional space, such that for a vocabulary $V = \{t_1, ..., t_N\}$, $\exists \mathbf{v}^{(i)} \in R^d, \forall i \in \{1, ..., |V|\}$. There are various techniques to construct these vectors, and the resulting embedding captures the meaning, context, and relationships between words.

In ADR-Mine, the word embeddings are created using the widely recognized *Word2Vec* (Mikolov et al., 2013). This family of algorithms trains shallow neural networks to produce word embeddings. The two primary methods for creating these embeddings are bag-of-words and skip-grams. The latter specifically involves predicting the context surrounding a particular token and was the approach used in the ADR-Mine paper.

Word2Vec attempts to capture all the possible meanings of a word in a single vector, which in-

evitably ignores the contextual nuances of each token. To address this limitation, contextualized embeddings have been developed using various techniques, with ELMo and BERT being among the most popular. BERT, with its larger number of parameters, offers higher generalization power than ELMo. Empirical studies have shown that BERT outperforms ELMo (Si et al., 2019). Therefore, we have opted to use pre-trained BERT to generate our embeddings.

For a given set of tokens $\{t_1, ..., t_N\}$, BERT produces a set of hidden activations $\{h_i^{(1)}, ..., h_i^{(l)}\}, \forall i \in N$. These constitute high-dimensional representations for each of the tokens and can therefore be considered to be embeddings in themselves. Studies have suggested that the earlier layers tend to capture lower-level linguistic features, while the last layers tend to capture more complex semantic and contextual features (Tenney et al., 2019). As such, the last layers are expected to be more informative when constructing embeddings.

There is a lack of agreement on the most effective method of combining the hidden activations of BERT. Different approaches have been proposed, including using the last layer, concatenating the last four layers or summing them (Alsentzer et al., 2019; Kazameini et al., 2020). In light of the curse of dimensionality faced by the downstream task of applying K-Means, the most suitable method is the latter, as it minimises dimensionality whilst providing richer contextual information.

The embeddings generated by BERT are contextualised, hence distinct for the same token in different contexts (Ethayarajh, 2019). However, since the model cannot be fine-tuned using unlabelled data, the only way to extract information from it is to average the distinct word embeddings for each unique token across the unlabelled dataset. It's important to note that ADRs typically do not have multiple meanings, and thus we expect the embeddings for each unique token to be similar. Hence, we define the embedding of a token t_i as:

$$\mathbf{v}^{(i)} = \frac{1}{M} \sum_{j=1}^{M} \sum_{k=l-4}^{l} h_{i,j}^{(k)}, \qquad (1)$$

where M is the number of times the token t_i appears in the dataset.

Clustering word embeddings

Given a set of vectors $\{\mathbf{v}^{(i)}\}_{i=1}^{N}$ the goal of the clustering problem is to partition the set into a number of subsets such that a given similarity measure (denoted by $D(\mathbf{v}^{(i)}, \mathbf{v}^{(j)})$), is greater within subsets than across them.

In ADR-Mine, the clustering method chosen was KMeans. KMeans finds the desired partition by initially assigning each vector to a random cluster and computing the corresponding cluster centroids. Then, it iteratively allocates each vector to the cluster whose centroid is nearest and updates the resulting centroids until convergence. The nearest centroid is determined by the choice of the similarity measure. In this case, Euclidean distance was chosen $D(\mathbf{v}^{(i)}, \mathbf{v}^{(j)}) = \sqrt{\sum_{k=1}^{d} (v_k^{(i)} - v_k^{(j)})^2}$. Upon convergence, KMeans returns a disjoint partition of the original set $S_1, S_2, ...S_K$, which in the case of embeddings, corresponds to K non-overlapping groups of words.

CRF for semantic segmentation

Conditional Random Fields (CRFs) are a type of statistical model used to represent the correlation between a sequence of observations and a sequence of labels. In the context of sequence labelling, the observations typically include the tokens located before and after the target token, as well as some supplementary attributes, such as whether the word is capitalised or negated (Zhang and Johnson, 2003; Lafferty et al., 2001). The goal is then to classify each individual token according to a set of predetermined classes. This classification requires annotated reviews.

In ADR-Mine, multiple input features were explored for each of the preprocessed string tokens. Ultimately, the most successful combination of features for each token t_i included:

- The neighbouring tokens: $\{t_j | j \in \{i-3, i-2, ...i+3\}\}$.
- The cluster assignments of the neighbouring tokens: $\{k | \mathbf{v}^{(j)} \in S_k, \forall j \in \{i-3, i-2, ...i+3\}\}$, where the partitions S_k are given by K-Means.

Each of the tokens was labelled as one of the following classes: *B-ADR, I-ADR, B-Indication, I-Indication,* and *Out*. When reproducing this model,

the classification was restricted to binary labels indicating whether the individual token was an ADR.

3.4 Fine-tuning BERT for sequence labelling

The BERT model for token classification is the second model applied, selected for its capability in comprehending the context of words within a sentence. As the task involves labelling each word in a sequence, the context of the word holds significant importance in identifying its label. In addition, BERT operates as a bidirectional model, allowing it to draw upon context from both directions of a sentence, facilitating more effective identification of word dependencies and context (Devlin et al., 2018).

The BERT model is pre-trained on large amounts of data which enables it to understand context very well. (Devlin et al., 2018) This allows it to perform well on tasks such as sequence labelling. The model is, thus, fine-tuned by training it on 70% of the available supervised data.

4 Experiments

4.1 Information Retrieval

For the BM25, the `BM25Okapi` library from `rank_bm25` is employed. The ADRs are preprocessed and fed to the model to serve as the document corpus. All the preprocessed reviews are looped through and the `get_scores()` function scores the ADRs based on their relevance to the review. The reviews with $score$=0 are removed and the remaining ones are normalized. Then, the ADRs above $threshold$=0.8 are considered relevant. The threshold was chosen such that the average number of ADRs retrieved is 5. The performance can be checked by comparing the result to the real ADRs. The final performance is obtained by averaging the precision, recall and f1-scores of all reviews.

The data does not need to be preprocessed before employing the TF-IDF, since it has its own tokenizer integrated into `TfidfVectorizer()`. The TF-IDF matrix is computed first by passing the ADR_lexicon list through the `fit_transform()` function. Each review is transformed using `transform()` and the `cosine_similarity()` is used as the scoring metric. The relevant ADRs are retrieved as above, using $threshold = 0.32$ and the performance metrics are found in the same way.

Both methods are then integrated within a newly created function named get_scores which receives as input the model used for the retrieval ('bm25' or 'tf_idf'), the ADRs, the reviews and the threshold for the chosen model.

4.2 Reproducing ADR-Mine

The word embeddings were obtained by implementing the skip-gram algorithm in the `word2vec` package of the `Gensim` python library. The vector size was set to 150 as in the original paper. The embeddings were created from the unlabelled data, which contained 216311 reviews and 17877934 tokens after lemmatization, resulting in a vocabulary of size 74157. The word embedding extraction took approximately 2 minutes with a CPU, resulting in 74157 embeddings for each of the tokens in the vocabulary.

The resulting vectors were clustered using the `sklearn` library. It implements the *kmeans++* algorithm, which only differs from the one described in section 3.3 in the initial cluster assignment. The number of clusters was set to $k = 150$, as in the original paper. The clustering took 8 minutes with a CPU, resulting in a cluster assignment for each of the unique tokens in the unlabelled dataset.

For the CRF sequence labelling task, the labelled data set was split into train, validation and test sets, with 1606, 200 and 206 reviews respectively. The CRF was implemented using the `sklearn_crfsuite` library. More specifically, the CRF was trained using the L-BFGS method. To avoid overfitting, early stopping was implemented, and the model was trained for 70 epochs, which took approximately 10 seconds with a CPU.

4.3 ADR-Mine with BERT embeddings

The BERT embeddings were extracted using the `transformers` library developed by Hugging Face. To start, we tokenized the $216,311$ reviews from the UCI ML Drug Review Dataset using the BERT Tokenizer. We then extracted the embeddings by computing the forward pass of the pre-trained BERT-base-uncased model on each batch of 100 reviews, storing the activations of the last 4 hidden layers. For each unique token, there are as many different embeddings as instances in the dataset. To

obtain a single embedding for each token, we took the mean of all corresponding embeddings across all instances. In the end, we obtained 19140 unique tokens, each with a corresponding 768-dimensional vector representation. The entire embedding extraction process took approximately 3 hours on a GPU.

The clustering was performed exactly as in the previous section, and the hyperparameter k was set to 800 using the Elbow method. Similarly, the sequence labelling process was identical to the previous section, except for the tokenization of the labelled dataset. In this case, we used the BERT Tokenizer for tokenization, rather than simply lemmatizing the text.

4.4 Fine-tuned BERT Model for Token Classification

Input Features

The dataset used for the BERT model implementation is the combined dataset of the CADEC, PsyTAR, and ADR datasets. The text of the ADR and the reviews are tokenized using the BERT Tokenizer. The input features used for this model are defined as follows:

- Token IDs - a tensor which contains the IDs associated with the respective token that maps to BERT's vocabulary.
- Attention mask - a tensor that contains 0s for padded tokens and 1s otherwise. This tensor is used to signal to BERT which parts of the tokens to pay attention to.
- Labels - the labels of the model where 1s signifies the position of an ADR, 0 signifies non-ADR tokens, and 2 specifies the padding.

All of the input features are padded to the maximum length of the longest tokenized review. Initially, there were 7 reviews which exceeded BERT's maximum length of 512 tokens. These reviews are split in half. After this step, the maximum length of the longest review becomes 506. All of the input features are padded to length 506.

After the long reviews are split, the total supervised data available consists of 2,162 rows.

Model Set-up

The model used is the BERT model for named entity recognition, more specifically, BertForTokenClassification, which is imported from the transformers library.

The number of classes is set to 3 - one class for the ADR tokens, one class for the non-ADR tokens, and one class for the paddings.

The model is trained on 70% of the available data, or on 1518 of the data points, and is tested on the remaining 30% of the data, or on 651 of the data points.

The Adam optimiser is used and the learning rate is set to 3e-5. The model is trained over 4 epochs. These hyperparameters are chosen according to (Sun et al., 2019) to minimise the risk of catastrophic forgetting of the pre-trained weights, i.e. to avoid over-fine-tuning the model and losing its ability to perform on out-of-sample data.

5 Results

The models are compared using four metrics: Precision, Recall, F1 Score, and Accuracy. The metrics are defined as follows:

$$p = \frac{tp}{tp + fp} \quad r = \frac{tp}{tp + fn} \quad f1 = \frac{2 * p * r}{p + r}$$

where p is precision, r is recall, $f1$ is f1 score, tp is true positives, fp is false positives, and fn is false negatives. The accuracy is simply the correctly labelled data over the overall data.

It's important to note that when calculating these metrics for the information retrieval models, specifically the TF-IDF and the BM25 models, the precision represents the number of relevant documents retrieved divided by the total number of documents retrieved, recall represents the number of relevant documents retrieved divided by the total number of relevant documents and accuracy is computed as the correctly classified documents (both relevant and irrelevant) divided by the total number of documents.

To ensure more accurate metric scores, the fine-tuned BERT model's metrics are computed by excluding padding from the ground-truth and predicted sequence labels and setting the positive label to '1', which denotes an ADR.

The performance for each model on these metrics is summarised in Table 1.

The results obtained from BM25 and TF-IDF are sub-optimal. The models cannot distinguish between the symptoms people had before taking the drug and afterwards. Also, they cannot identify the negations in the sentence, so for structures like

Model	Precision	Recall	F1 Score	Accuracy
TF-IDF	0.053	0.066	0.046	-
BM25	0.124	0.092	0.090	-
ADRMine	0.670	0.585	0.625	0.893
ADRMine with BERT	0.753	0.663	0.705	0.916
Fine-tuned BERT	0.778	0.757	0.767	0.934

Table 1: Comparison of ADR extraction across models using Precision, Recall, F1 Score, and Accuracy. * Note that accuracy was not calculated for the BM25 and TF-IDF as it is not a representative measure for the information retrieval models.

'My stomach does not hurt anymore', the models would return 'stomach ache'. That being said, an improvement can still be observed from the performance of the TF-IDF to the one of the BM25: from 0.053 precision and 0.066 recall to 0.124 and 0.092, respectively.

The ADRMine model outperforms the information retrieval models by a large margin as observed by the F1 Score. This stands to show that making use of Word2Vec embeddings, a larger unsupervised data corpus, and a machine learning model could bring about an improvement in ADR extraction.

The results of this study indicate that the proposed ADRMine model which utilizes BERT, and the fine-tuned BERT, exhibit superior performance compared to both information retrieval methods and the ADRMine model. This highlights the effectiveness of transformer models and their valuable contribution to ADR extraction and sequence labelling tasks.

6 Conclusion

A novel transformers-based approach for sequence labelling in relation to ADR identification was introduced in this paper. It was shown to outperform established data-mining baselines in every regard. An attempt at improving the ADR-Mine model was successful, with a significantly superior performance thanks to the contextualised embeddings pre-trained on unannotated data. The results obtained represented improvements over the previous literature and transformers are currently the state-of-the-art architecture for NLP tasks. For these reasons, they are likely to be the best-suited architecture for models aiming to identify drug effects in unstructured data, at least in the foreseeable future.

The implications of further developing reliable and robust models for this purpose could, potentially, be of great benefit to both pharmaceutical companies and regulatory bodies in charge of drug commercialisation and supervision. The large-scale data nowadays available offers a great opportunity to complement current post-market research with first-hand reports from the patients themselves, which if aggregated and analysed may reveal some previously undiscovered patterns. These could range from identifying adverse reactions that were previously unseen or whose frequency was underestimated, to insights on drug interactions or even potential beneficial effects.

Nevertheless, it remains too early to predict what the impacts could be: the results are promising but far from what would be necessary to roll out a model ready for real-world application. Future research could involve the development of superior domain-specific models. Examples of this could be fine-tunings of alternative architectures such as BioBERT, a hybrid architecture which combines the BERT model with a conditional random field (CRF) layer to perform NER on biomedical text (Lee et al. (2020)). Other efforts could be focused on developing successful and less costly pseudo-annotation methods for the large quantity of unlabelled data which is currently available, to take full advantage of its potential. Previous attempts, such as in Yates et al. (2015) where this challenge was addressed through an alternative annotation scheme involving ADRs listed on drug labels, have not produced convincing improvements.

References

Emily Alsentzer, John R. Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew B. A. McDermott. 2019. Publicly available clinical bert embeddings.

Jiang Bian, Umit Topaloglu, and Fan Yu. 2012. Towards large-scale twitter mining for drug-related adverse events. In *Proceedings of the 2012 international workshop on Smart health and wellbeing*, pages 25–32.

Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic Acids Res.*, 32(Database-Issue):267–270.

Delroy Cameron, Gary A. Smith, Raminta Daniulaityte, Amit P. Sheth, Drashti Dave, Lu Chen, Gaurish Anand, Robert Carlson, Kera Z. Watkins, and Russel Falck. 2013. Predose: A semantic web platform for drug abuse epidemiology using social media. *Journal of Biomedical Informatics*, 46(6):985–997. Special Section: Social Media Environments.

Cristóbal Colón-Ruiz and Isabel Segura-Bedmar. 2020. Comparing deep learning architectures for sentiment analysis on drug reviews. *Journal of Biomedical Informatics*, 110:103539.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Thu Dinh and Goutam Chakraborty. 2020. Detecting side effects and evaluating the effectiveness of drugs from customers' online reviews using text analytics, sentiment analysis, and machine learning models. In *sas-global-forum-proceedings*, pages 1–23.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings.

Vinodhini Gopalakrishnan and Chandrasekaran Ramaswamy. 2017. Patient opinion mining to analyze drugs satisfaction using supervised learning. *Journal of Applied Research and Technology*, 15(4):311–319.

Felix Gräßer, Surya Kallumadi, Hagen Malberg, and Sebastian Zaunseder. 2018. Aspect-based sentiment analysis of drug reviews applying cross-domain and cross-data learning. In *Proceedings of the 2018 International Conference on Digital Health (DH '18)*, pages 121–125, New York, NY, USA. ACM.

Sarvnaz Karimi, Alejandro Metke-Jimenez, Madonna Kemp, and Chen Wang. 2015a. Cadec: A corpus of adverse drug event annotations. *J Biomed Inform*, 55:73–81.

Sarvnaz Karimi, Chen Wang, Alejandro Metke-Jimenez, Raj Gaire, and Cecile Paris. 2015b. Text and data mining techniques in adverse drug reaction detection. *ACM Comput. Surv.*, 47(4).

Amirmohammad Kazameini, Samin Fatehi, Yash Mehta, Sauleh Eetemadi, and Erik Cambria. 2020. Personality trait detection using bagged svm over bert word embedding ensembles.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML*, 1:282–289.

Robert Leaman, Laura Wojtulewicz, Ryan Sullivan, Annie Skariah, Jian Yang, and Graciela Gonzalez. 2010. Towards internet-age pharmacovigilance: extracting adverse drug reactions from user posts to health-related social networks. In *Proceedings of the 2010 workshop on biomedical natural language processing*, pages 117–125.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Jingjing Liu, Alice Li, and Stephanie Seneff. 2011. Automatic drug side effect discovery from online patient-submitted reviews: Focus on statin drugs. In *Proceedings of First International Conference on Advances in Information Mining and Management (IMMM), Barcelona, Spain*, pages 23–29.

Xiao Liu and Hsinchun Chen. 2013. Azdrugminer: an information extraction system for mining patient-reported adverse drug events in online patient forums. In *Smart Health: International Conference, ICSH 2013, Beijing, China, August 3-4, 2013. Proceedings*, pages 134–150. Springer.

Michelle Meadows. 2002. The fda's drug review process: ensuring drugs are safe and effective.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality.

Azadeh Nikfarjam, Abeed Sarker, Karen O'Connor, Rachel Ginn, and Graciela Gonzalez. 2015. Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. *Journal of the American Medical Informatics Association*, 22(3):671–681.

Yuqi Si, Jingqi Wang, Hua Xu, and Kirk Roberts. 2019. Enhancing clinical concept extraction with contextual embeddings. *Journal of the American Medical Informatics Association*, 26(11):1297–1304.

M. Stephens, D. Buckingham, J. C. Talbot, and P. Routledge. 1985. *The Detection of New Adverse Drug Reactions*. Stockton Press.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *Chinese Computational Linguistics: 18th China National Conference, CCL 2019, Kunming, China,*

October 18–20, 2019, Proceedings 18, pages 194–206. Springer.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. Bert rediscovers the classical nlp pipeline.

Henry Tsai, Jason Riesa, Melvin Johnson, Naveen Arivazhagan, Xin Li, and Amelia Archer. 2019. Small and practical bert models for sequence labeling.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

JL. Wu, LC. Yu, and PC. Chang. 2012. Detecting causality from online psychiatric texts using inter-sentential language patterns. *BMC Medical Informatics and Decision Making*, 12(1):72.

Andrew Yates, Nazli Goharian, and Ophir Frieder. 2015. Extracting adverse drug reactions from social media. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.

Q. Zeng, S. Kogan, N. Ash, R. A. Greenes, and A. A. Boxwala. 2002. Characteristics of consumer terminology for health information retrieval. *Methods of Information in Medicine*, 41(4):289–298.

Tong Zhang and David Johnson. 2003. A robust risk minimization based named entity recognition system. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, page 204–207, USA. Association for Computational Linguistics.

Maryam Zolnoori, Kin Wah Fung, Timothy B. Patrick, Paul Fontelo, Hadi Kharrazi, Anthony Faiola, Nilay D. Shah, Yi Shuan Shirley Wu, Christina E. Eldredge, Jake Luo, Mike Conway, Jiaxi Zhu, Soo Kyung Park, Kelly Xu, and Hamideh Moayyed. 2019. The psytar dataset: From patients generated narratives to a corpus of adverse drug events and effectiveness of psychiatric medications. *Data in Brief*, 24:103838.

Appendix

Cluster	Semantic Category	Examples of clustered words
c_1	Illness/Disease	migraine, diabetes, schizophrenia, epilepsy
c_2	Action Verb	sleep, walk, move, sit, breathe
c_3	Directions/Locations	up, out, back, down, away
c_4	Medication	ibuprofen, tramadol, methadone, oxycodone
c_5	Symptoms	dizzy, anxious, drowsy, weak, nauseated

Table 2: Examples of the unsupervised learned clusters with the subsets of the words in each cluster using word2vec embeddings

Cluster	Semantic Category	Examples of clustered words
c_1	Emotions	fear, stress, blame, grief, tension
c_2	Injury	hurt, injury, burnt, destroyed
c_3	Dates/Order	25th, 50th, 30th, 24th, 26th
c_4	Body Parts	hands, feet, arms, lips, fingers
c_5	Body Systems	respiratory, artery, cardiac, nerve, vascular

Table 3: Examples of the unsupervised learned clusters with the subsets of the words in each cluster using BERT embeddings