# Evaluating Language Models for Adverse Drug Event Recognition on CADEC

Phillip Shen
x.shen.4@umail.leidenuniv.nl
Leiden University, LIACS
the Netherlands, Leiden

Xiangyu Li
x.li.55@umail.leidenuniv.nl
Leiden University, LIACS
the Netherlands, Leiden

## Abstract

Access to medical information through online platforms has increased significantly, creating a need for efficient processing of user-generated medical content. This study evaluates and compares the performance of various language models for Named Entity Recognition (NER) tasks in identifying adverse drug events using the CSIRO Adverse Drug Event Corpus (CADEC). We assess traditional approaches (Word2Vec) against modern transformer-based models (BERT, ELECTRA) and their domain-specific variants (BioCliBERT, ELECTRA-MED). Our experiments demonstrate that domain-specific models generally outperform their base versions, with BioCliBERT achieving the highest overall F1 score (0.646) after hyperparameter optimization. While modern models showed better performance in recognizing less frequent entity types like diseases and symptoms, traditional Word2Vec remained competitive for common entities like drug names and adverse effects. The results highlight the importance of domain adaptation and careful hyperparameter tuning in medical NER tasks, while also suggesting that simpler models should not be dismissed in resource-constrained settings.

## Keywords

Natural Language Processing, Named Entity Recognition, CADEC, Language Model, ELECTRA, ELECTRA-MED, BERT, BioCliBERT

## 1 Introduction

Access to healthcare resources, particularly General Practitioners (GPs), is essential for public well-being and is closely tied to life expectancy[3]. However, the growing population has outpaced the availability of GPs, creating a significant resource gap. The COVID-19 pandemic further highlighted this issue, leading to an increase in online medical consultations as a supplement to traditional care[13]. While these consultations have helped address the shortage, they also bring new challenges, especially in providing accurate and detailed prescription advice due to the absence of physical interaction. Patients now require more information about medications, including their intended use and potential adverse effects. To effectively process and analyze this increased demand for medication information, automated solutions are becoming essential.

Named Entity Recognition (NER) has emerged as a promising solution in the Natural Language Processing (NLP) domain. Currently, many automated methods for NER have already been implemented[1, 4, 7, 9]. By employing these techniques, automated medical information extraction can be made thus improving the effectiveness of online consultations, especially for adverse effects and symptoms. Our research intends to compare these state-of-the-art models' results on NER tasks in identifying and categorizing critical medical entities such as drug names, symptoms, and adverse events, using the CSIRO Adverse Drug Event Corpus (CADEC)[8].

Our study aims to answer these research questions:

- : What is the performance gap between different models?
- : How would clinical-domain-specific variations of modern models perform compared to their base version?
- : What adjustment can be done to improve NER results?

By evaluating these models using metrics such as precision, recall, and F1-score, we seek to identify the most effective approach for extracting medication-related information from patient narratives.

## 2 Related work

In this section we would expand on the preliminary knowledge of NLP trend and recent researches on NER.

In the last decade, the NLP field has experienced fundamental revolutions. Traditional NLP methodologies centered on manual feature engineering and probabilistic approaches, primarily focusing on shallow neural architectures[6, 10]. The giant revolution occurred in 2014 with the introduction of the attention mechanism, resulting in the widespread of deep neural architectures[2]. In 2017, Vaswani et al.[12] proposed the transformer architecture, introducing a novel self-attention mechanism to construct highly efficient encoder-decoder language models. This innovation established new state-of-the-art benchmarks across a large quantity of NLP tasks and laid the foundation for modern language models.

Particularly, NLP models now can learn a vast amount of data in various scenarios like text analysis and classification due to deep learning methods. Bidirectional Encoder Representations from Transformers (BERT) is one of the most famous model with transformer architecture[5]. It is a bidirectional encoding model that aims to generate a text representation by masking original tokens then reconstructing the masked tokens trough training. Different variations of BERT have been proposed for certain use, one example of which is the medical domain. BioBERT[11] is trained on biomedical literature, which is better at understanding medical contexts and BioCliBERT[1] is further fine-tuned from BioBERT, with specialized vocabulary focusing on clinical terms. BERT's architecture incorporates position embeddings and segment embeddings alongside traditional token embeddings, enabling it to capture both sequential relationships and sentence-pair relationships[5]. The self-attention mechanism allows the model to weigh the importance of different tokens in the input sequence dynamically, capturing long-range dependencies and complex patterns. Each transformer layer processes these attention-weighted representations through a feed-forward neural network, progressively building more complex representations. By stacking up 12 transformer layers, models from BERT family perform extremely well on NER tasks[9].

Based on transformer architecture, Clark et al.[4] proposed another model called Efficiently Learning an Encoder that Classifies Token Replacements Accurately (ELECTRA). ELECTRA uses a generator-discriminator architecture like Generative adversarial network where the generator, a small masked language model, produces plausible replacements for input tokens, while the discriminator learns to distinguish between original and replaced tokens. This approach is computationally more efficient as it trains on all input tokens rather than just the masked subset like in BERT models. ELECTRA's discriminative pre-training objective naturally aligns with token-level classification tasks of NER, as both involve binary discrimination at the token level - detecting replaced tokens during pre-training and identifying entity/non-entity tokens during NER tasks. This architectural alignment potentially contributes to ELECTRA's efficient learning of token-level features relevant to entity recognition. Additionally, like BERT family, ELECTRA also has various variation for different downstream tasks, ELECTRA-MED[7] being the one fine-tuned from medical corpus.

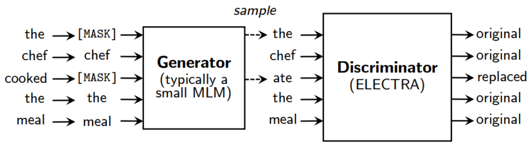

**Figure 1: Example of Replaced Token Detection[4]**

## 3 Data

### 3.1 Description

CADEC is a rich annotated corpus with medical information from online social media, a forum called AskaPatient. Annotations contain mentions of concepts such as drugs, adverse effects, symptoms, and diseases linked to their corresponding concepts in controlled vocabularies like SNOMED Clinical Terms and MedDRA[8]. Specifically, the original annotations in CADEC have 4 category tags, which are drugs, adverse effects, symptoms, and diseases. Drug annotates the name of a medicine or drug. Drug classes, such as Nonsteroidal anti-inflammatory drugs (NSAIDS), as well as medical devices were excluded. Adverse effects refer to the side effects that are strongly related to the drug being used. Disease is the reason that the patient takes some certain drug, and symptom is the biological outcome of the disease. After linking to third-part vocabularies, those tags are classified into more precise tags. For example, general adverse effects tags are replaced by specific events like hypoglycemia.

These posts contain patient demographics, a satisfaction rating on the medication from 1 (low) to 5 (high), reason for taking the medication, how it was administered, patient comments on the effectiveness of the drug and if any side effects were experienced. Only the free text sections of each post are annotated and provided in CADEC. Table 1 shows an example of the post.

CADEC dataset has twelve drugs in total, most of whose names are different from the chemical ingredient. Medications are in two categories: Diclofenac, which includes those medications with Diclofenac in their active ingredient, and Lipitor. Table 2 shows the mapping between drug names and active chemical ingredient and table 3 shows the class distribution in CADEC.

**Table 2: Drug name and ingredient mapping**

| Name | Ingredient |
|---|---|
| Voltaren | Diclofenac Sodium |
| Cataflam | Diclofenac Potassium |
| Voltaren-XR | Diclofenac Sodium |
| Arthrotec | Diclofenac Sodium; Misoprostol |
| Pennsaid | Diclofenac Sodium |
| Solaraze | Diclofenac Sodium |
| Flector | Diclofenac Epolamine |
| Cambia | Diclofenac Potassium |
| Zipsor | Diclofenac Potassium |
| Diclofenac Sodium | Diclofenac Sodium |
| Diclofenac Potassium | Diclofenac Potassium |
| Lipitor | Atorvastatin Calcium |

**Table 3: Entity Type Distribution per Drug in CADEC**

| Drug Name | ADR | Disease | Drug | Symptom |
|---|---|---|---|---|
| Voltaren | 145 | 19 | 41 | 34 |
| Cataflam | 21 | 3 | 17 | 10 |
| Voltaren-XR | 50 | 3 | 16 | 3 |
| Arthrotec | 567 | 24 | 149 | 140 |
| Pennsaid | 5 | 0 | 3 | 4 |
| Solaraze | 5 | 1 | 2 | 2 |
| Flector | 0 | 1 | 1 | 1 |
| Cambia | 6 | 3 | 3 | 0 |
| Zipsor | 4 | 0 | 0 | 3 |
| Diclofenac Sodium | 34 | 1 | 8 | 9 |
| Diclofenac Potassium | 4 | 0 | 0 | 0 |
| Lipitor | 4475 | 223 | 1552 | 34 |

For NER task in our research, we put the emphasis on tokenization and data tagging. The majority of text in CADEC is largely written in colloquial language and often deviates from formal English grammar and punctuation rules, raising up the difficulty to make use of the data for NER tasks. Even though the dataset is based on online resources, it was before the era of widespread of auto-correction, resulting in a large quantity of typographical errors like 'combined' misspelled as 'comboned'. Meanwhile, since the data is extracted from online forum, some patients kept their forum habit when writing down paragraphs, which led to several use of Kaomoji like ':/ ' for being helpless and 'D:' for being sad. Those noise data might lead to potential issues in tokenization.

### 3.2 Exploration

There are in total 1253 posts (7398 sentences) in CADEC dataset. Table 4 shows general distribution of the corpus as well as for the two categories. The posts with text are our research focus and it is

**Table 1: A sample post on AskaPatient.com**

| Rating | Reason | Side effects | Comments | Sex | Age | Duration/dosage | Date |
|---|---|---|---|---|---|---|---|
| 4 | Osteoarthritis of the hip | It helps relieve chronic pain but over time, causes intestinal pain and bleeding. I had symptoms similar to diverticulitis:<br>blood in stool, pain. | I would be cautious about paying attention to cramping, intestinal /stomach pain which can lead to very serious conditions. | M | 63 | 1.5 years<br>100MGER1XD | 4/12/ 2013 |

where the NER task would be experimented. Figure 2 shows high frequency words in CADEC corpus. With more than 1000 posts and much outnumbering Diclofenac category, Liptor is the main charatacter in the word cloud. Additionally, muscle pain is the focus in the corpus for being the major side effect of Liptor.

**Table 4: Statistics on the data used in CADEC.**

| | Corpus | Diclofenac | Lipitor |
|---|---|---|---|
| No. posts | 1321 | 264 | 1057 |
| No. posts with text | 1250 | 250 | 1000 |
| No. sentences | 7632 | 1263 | 6369 |
| Avg. post length (sentence) | 6 | 5 | 6 |
| No. words | 101,486 | 16,778 | 84,708 |
| Avg. post length (word) | 81 | 67 | 85 |
| Time span | Jan 2001–Sep 2013 | Feb 2002–Aug 2013 | Jan 2001–Sep 2013 |
| Gender:F | 662 (50.1%) | 181 (68.6%) | 481 (45.6%) |
| Gender:M | 617 (49.9%) | 76 (28.8%) | 541 (51.2%) |
| Age range | 17–84 | 17–78 | 19–84 |
| Avg. age | 52 | 47 | 54 |

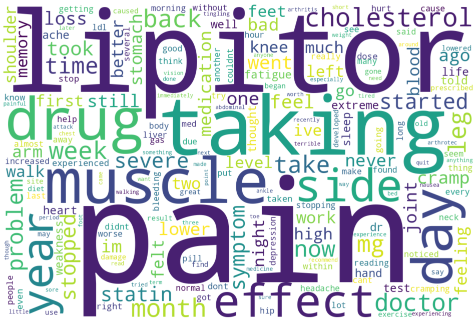

**Figure 2: Wordcloud for CADEC**

## 4 Methods

The models have been implemented by using the Python programming language. Particularly, we used the Pytorch syntax and the Transformer Huggingface library. The code has been run on an MSI Vector 16 laptop with 64GB of RAM, an Intel i9-13980HX CPU with 24 cores, an RTX 4080 Laptop GPU with 12GB dedicated GPU memory and 32GB shared memory.

### 4.1 Preprocessing

In this section we would introduce how we made the IOB file for NER.

When it comes to assigning semantic labels to text segments in order to make an IOB format file, the original annotations along with two sets that are linked to SNOMED or MedDRA are all available. Since we focus on the models' performance on the 4 main tags rather than specifying between detailed adverse effects or diseases, we decided to choose the original annotations. Simply put, we expect the IOB file to link the human-made labels in the annotation with the text from the forum's posts.

Many symptoms and adverse effects include several stop words like 'a bit of', so in our study we do not exclude stop words in the dataset. After wiping out non-text posts, we link the tags from annotation file in the original folder with the text files in the text folder for each corresponding file and successfully extracted the IOB file in tsv format as we previously intended. The paragraph number is 1250, aligning with the number in Table 4.

For model training, we split the whole corpus in the ratio 8:1:1 and created 3 different datasets which are train, test and validation set for cross-validation and hyperparameter optimization. There are 998 examples in the train set, 124 in test set and 126 in validation set. The random seed used in the split is 42 for reproduction.

### 4.2 Model Application

In this section we would expand on the models we would use and the evaluation metrics on NER results.

The baseline we choose is a traditional Word2Vec model as a comparison against state-of-the-art models. The metric we use for evaluations are recall, precision and F1 score which are commonly used. Additionally, 2 encoders from BERT and BioCLiBERT with weights along with 2 discriminators from ELECTRA and ELECTRA-MED with weights are trained.

For the simplicity of the results we made a mapping between original annotated tags and the abbreviations we adopted in our research shown in Table 5.

**Table 5: Abbreviation and original tags mapping**

| Original Tag | Abbr |
|---|---|
| Adverse Drug Effect | ADR |
| Drug | DRU |
| Disease | DIS |
| Symptom | SYM |

#### 4.2.1 Word2Vec

Table 6 shows the parameter used in Word2Vec model and the implementation of Word2Vec is based on Gensim's Python library. This baseline is mainly for comparison between old cliche model and state-of-the-art models, as a result we do not further explore the result of Word2Vec.

**Table 6: Hyperparameter Configuration for Word2Vec Model**

| Parameter | Value |
|---|---|
| Embedding Dimension | 100 |
| Hidden Dimension | 128 |
| Batch Size | 32 |
| Training Epochs | 10 |
| Window Size | 5 |
| Min Count | 1 |
| Skip-Gram | 1 |

#### 4.2.2 BERT and BioCliBERT

To capture the sequential context, BERT model is adopted, along with one variation in medical domain called BioCliBERT for the exploration of how the domain specified model would perform compared with the base model in our task. For the basic BERT model[5], we used the tokenizer and trainer from Google's bert-base-uncased model on HuggingFace, which has an an uncased English vocabulary along with 12 layers, 768 hidden layers, 12 heads and 168M parameters. For BioCliBERT, we used the tokenizer and trainer from emilyalsentzer/Bio_ClinicalBERT on HuggingFace, which is initialized with BERT-Base and trained on all notes from MIMIC III, a database containing electronic health records from ICU patients at the Beth Israel Hospital in Boston[11]. The hyperparameter we used is listed in Table 7.

**Table 7: Hyperparameter Configuration for BERT**

| Parameter | Value |
|---|---|
| Max Length | 512 |
| Learning Rate | 2e-5 |
| Batch Size | 16 |
| Epoch | 3 |
| Optimizer | Adam |

#### 4.2.3 ELECTRA and ELECTRA-MED

Another branch of transformer model we take into account is the ELECTRA. Similar to GAN, ELECTRA has a generator and a discriminator, the former to replace tokens and the latter to learn from training and to gain the capability of distinguishing between replaced tokens and original tokens. Since the essence of the discriminator is very close to NER task, we consider ELECTRA as a very powerful model to do NER task. Like the BERT part in previous section, we implemented the base ELECTRA and one of its variation in medical domain. The base ELECTRA model we used is from google/electra-small-discriminator on HuggingFace and ELECTRA-MED is from kamalkraj/bioelectra-base-discriminator-pubmed with the full training corpus. The hyperparameter used in the first round is listed in Table 8. Since ELECTRA is computational-light, we used a set of more extensive hyperparameters containing more training process.

**Table 8: Hyperparameter Configuration for ELECTRA**

| Parameter | Value |
|---|---|
| Max Length | 512 |
| Learning Rate | 5e-5 |
| Batch Size | 16 |
| Epoch | 10 |
| Optimizer | Adam |
| Generator Loss Weight | 50 |
| Discriminator Loss Weight | 1 |
| Masked Token Percentage | 15% |

#### 4.2.4 Evaluation

The evaluation metric we use is precision, recall and F1 score. Precision is the fraction of relevant items among the retrieved items, and recall is the fraction of relevant items that were retrieved, both based on relevance. F1 score represents a harmonious balance between precision and recall, computed as their harmonic mean. Specifically, high precision reduces false positive burden while high recall ensures critical entities aren't missed while F1 reflects the overall performance of the model on all aspects.

## 5 Results

### 5.1 Baseline: Word2Vec

Table 9 shows the result of Word2Vec model which we used as baseline. The training runtime is 62 seconds using CPU resource. The evaluation scores for ADR tag are all around 0.55, which is relatively satisfactory. The DRU class has an astonishingly high precision score, recall and F1 being very effective as well. For DIS and SYM, the detected tag numbers are both lower than 100 and the precision for both tags are no more than 0.2, indicating large quantity of false positives, and low recalls stand for the model missing significant entities. For the O labels, the model performed very well as each score is more than 0.9.

**Table 9: Word2Vec results**

| Label | Precision | Recall | F1-Score | Number |
|---|---|---|---|---|
| ADR | 0.540 | 0.568 | 0.553 | 1078 |
| DRU | 0.902 | 0.769 | 0.830 | 143 |
| DIS | 0.174 | 0.342 | 0.230 | 38 |
| SYM | 0.134 | 0.223 | 0.163 | 67 |
| O | 0.958 | 0.946 | 0.952 | 9868 |

## 5.2 BERT, ELECTRA and variations

For the rest models, performance of O labels are all as high as above 95% so for presentation convenience the emphasis is set on the main 4 tags of ADR, DRU, DIS and SYM.

Table 10 shows the results for BERT and BioCliBERT. The BERT training runtime is around 70 seconds, approximately taking up the same time as Word2Vec due to GPU acceleration, but neither of the base BERT nor BioCliBERT outperformed Word2Vec on all aspects. Besides, for SYM tag both BERT models failed to make any right predicition, rather the Word2Vec got some correct predictions even though the precision and recall are quite low. BERT models achieved a slightly worse performance on ADR and DRU classes, indicating that BERT might need further training. Another potential factor leading to this might be that the contextual understanding of transformer architecture is not extremely helpful at detecting medical classes because on one hand most patients are not familiar with accurate medical expressions and on the other many symptoms and diseases would just occur out of expectation, making the long term memory of the model less helpful. Letting other things alone, the medical domain specified BioCliBERT model outperforming the base BERT proved that for medical NER task, it would be more useful to fine-tune the model than solely using a state-of-the-art model.

**Table 10: BERT (top) and BioCliBERT (bottom) results**

| Label | Precision | Recall | F1-Score | Number |
|---|---|---|---|---|
| ADR | 0.416 | 0.500 | 0.454 | 665 |
| DRU | 0.814 | 0.857 | 0.835 | 400 |
| DIS | 0.000 | 0.000 | 0.000 | 58 |
| SYM | 0.000 | 0.000 | 0.000 | 50 |
| **Overall** | 0.554 | 0.576 | 0.565 | 1173 |
| **Label** | **Precision** | **Recall** | **F1-Score** | **Number** |
| ADR | 0.465 | 0.540 | 0.500 | 607 |
| DRU | 0.863 | 0.885 | 0.874 | 322 |
| DIS | 0.217 | 0.086 | 0.123 | 58 |
| SYM | 0.000 | 0.000 | 0.000 | 50 |
| **Overall** | 0.584 | 0.595 | 0.590 | 1037 |

Table 11 shows the result of ELECTRA models.The overall training runtime is around 50 seconds but with more computational-heavy hyperparameters. With more training steps the results are much better than BERT for ADR and DRU, outperforming Word2Vec slightly. For DIS and SYM, there are satisfactory improvements compared with Word2Vec and BERTs on precision, indicating the transformer architecture is able to make true positive predictions with enough training. For ELECTRA-MED, it also has better results than ELECTRA with approximately 5% of improvement, aligning with the improvement shown in BERT when fine-tuning the model on a medical corpus.

**Table 11: ELECTRA (top) and ELECTRA-MED (bottom) results**

| Label | Precision | Recall | F1-Score | Number |
|---|---|---|---|---|
| ADR | 0.506 | 0.569 | 0.535 | 571 |
| DRU | 0.917 | 0.814 | 0.862 | 301 |
| DIS | 0.254 | 0.250 | 0.252 | 52 |
| SYM | 0.400 | 0.041 | 0.075 | 48 |
| **Overall** | 0.606 | 0.601 | 0.604 | 972 |
| **Label** | **Precision** | **Recall** | **F1-Score** | **Number** |
| ADR | 0.520 | 0.588 | 0.552 | 571 |
| DRU | 0.913 | 0.804 | 0.855 | 301 |
| DIS | 0.285 | 0.307 | 0.296 | 52 |
| SYM | 0.384 | 0.104 | 0.163 | 48 |
| **Overall** | 0.611 | 0.616 | 0.614 | 972 |

## 5.3 Hyperparameter Optimization

After the first round, we decided to do a Hyperparameter Optimization (HPO) for BERTs since they failed to deliver a state-of-the-art model's performance and we assumed the hyperparameters we used is not sufficient for the model to comprehensively learn the corpus, considering with extensive hyperparameters ELECTRAs got a much better results. To see whether the results of BERTs can be improved or not, we used grid search on the validation set, whose details are in Table 12. F1 score is considered as evaluation metric to seek for the best configuration combination.

**Table 12: Hyperparameter Grid for BERTS**

| Hyperparameter | Values |
|---|---|
| Learning Rate | 1e-5, 2e-5, 5e-5 |
| Batch Size | 8, 16 |
| Training Epoch(s) | 3, 6 |

### 5.3.1 Base BERT

Figure 3 shows the HPO results for base BERT model, the combination with highest F1 score is learning rate 5e-5, batch size 16 and epoch 6.

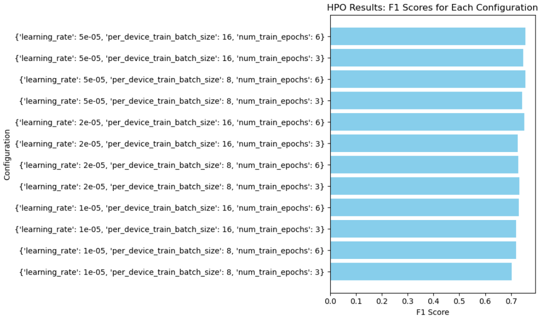

**Figure 3: Base BERT HPO F1 results**

Table 13 shows the NER results using the best hyperparameter combination found in HPO for BERT. ADR class had a significant improvement around 20% for precision and F1 score, 10% for recall. For DRU the precision also got improved by 10%. FOr DIS and SYM, two classes that base BERT initially failed to correctly detect any instance, got a better result than Word2Vec and the F1 score for SYM is even higher than ELECTRA-MED. Additionally, we merely used 3 and 6 as training epoch, both smaller than 10 which is used in Word2Vec training, demonstrating that proper hyperparameter tuning can substantially improve NER performance across different entity types for modern models like BERT.

**Table 13: Base BERT NER results after HPO**

| Label | Precision | Recall | F1-Score | Number |
|---|---|---|---|---|
| ADR | 0.555 | 0.548 | 0.552 | 665 |
| DRU | 0.925 | 0.870 | 0.896 | 400 |
| DIS | 0.255 | 0.206 | 0.228 | 58 |
| SYM | 0.473 | 0.180 | 0.260 | 50 |
| **Overall** | 0.667 | 0.625 | 0.645 | 1173 |

### 5.3.2 BioCliBERT

Figure 4 shows the HPO results for BioCliBERT model, the combination with highest F1 score is learning rate 5e-5, batch size 16 and epoch 3.

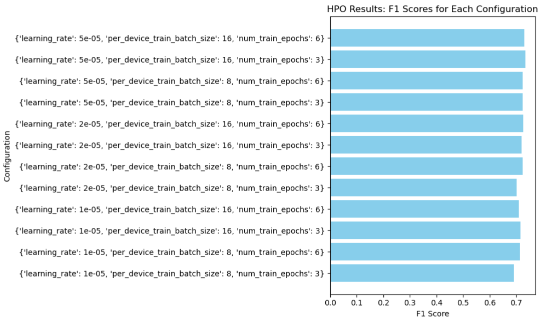

**Figure 4: BioCliBERT HPO F1 results**

Table 14 shows the NER results using the best hyperparameter combination found in HPO for BioCliBERT. Like what happened to base BERT, BioCliBERT reached a new peak performance as well, higher socres on ADR and DRU, non 0 and high scores on DIS and SYM, becoming the best result in our research. All 4 classes got significantly improvement and all the F1 scores are the highest among all other cases. What is out of expectation is the epoch being used is 3 rather than 6. Potential factor leads to this might be the selection metric being F1 so subtle details are neglected and there is no need to train the model with too many epochs. Overall, the result for BioCliBERT after HPO is truly satisfactory, accomplishing the intended NER task we intended to finish.

**Table 14: BioCliBERT NER results after HPO**

| Label | Precision | Recall | F1-Score | Number |
|---|---|---|---|---|
| ADR | 0.543 | 0.571 | 0.557 | 607 |
| DRU | 0.906 | 0.931 | 0.918 | 322 |
| DIS | 0.323 | 0.362 | 0.341 | 58 |
| SYM | 0.343 | 0.220 | 0.268 | 50 |
| **Overall** | 0.636 | 0.654 | 0.646 | 1037 |

## 6 Discussion

### 6.1 Performance Gap Between Models

The experimental results demonstrate a clear progression in model performance, with domain-specific models generally outperforming their bases. Most notably, BioCliBERT with optimized hyperparameters achieved the highest overall F1 score (0.646), surpassing both the baseline Word2Vec model and other transformer models. This improvement was particularly evident in the recognition of ADR and DRU, where BioCliBERT achieved F1 scores of 0.918 and 0.557 respectively.

Modern models have the potential to learn much more than traditional models in fewer training steps. When Word2Vec is trained on 10 epochs, it still achieved relatively low scores on DIS and SYM due small quantity of data but merely with up to 6 epochs the result from BERT is much better, letting alone other advantages and with the same epoch of 10, ELECTRA achieved overall better results than Word2Vec without further optimization.

### 6.2 Impact of Domain Specialization

The better performance of domain-specific models (BioCliBERT and ELECTRA-MED) over their base versions (BERT and ELECTRA) indicates the significance of domain adaptation in medical NER tasks. This advantage can be attributed to their pre-training on medical corpora, which offers a better understanding of medical terminology and context.

### 6.3 Hyperparameter Optimization Effects

Our HPO experiments revealed that careful tuning can substantially improve model performance, particularly for BERT-based models. The most significant improvements were observed with learning rates around 5e-5 and moderate batch sizes which in our case is 16, suggesting these parameters are crucial for balancing learning stability and effectiveness.

### 6.4 Limitations

We tried to carried out as extensive comparisons as possible but there are some less significant parts are missed. We assumed Word2Vec would not improve as much as modern models so we didn't do further training but with accurate data to support the assumption would reduce the arbitrariness and make our research more robust. We put our main focus on BERTs, and ELECTRAs were left in the first stage. Even though the results of ELECTRA models were satisfactory enough it still remains unknown whether they could outperform BERTs after tuning.

For the dataset, the distribution of CADEC is not even, both in the total number of entities across drugs and the distribution of entity types. Lipitor disproportionately contributes to the dataset, while many other drugs have much fewer representation, potentially biasing predictions toward drugs with higher representation and entity types like ADRs, which are more frequent.

## 7 Conclusion

Our study provides a comprehensive evaluation of modern language models for adverse drug event recognition in user-generated medical content.

Traditional models are not good at detecting classes with few sample data, like DIS and SYM in our research, but with enough training modern models achieved much better results. The results demonstrate that domain-specific pre-training and careful hyperparameter optimization are crucial for achieving optimal performance in medical NER tasks. The superior performance of BioCliBERT and ELECTRA-MED over their base versions validates the importance of domain adaptation in medical NLP applications. However, the strong performance of the Word2Vec baseline on certain tasks like ADR and DRU recognition suggests that simpler models should not be dismissed without consideration, particularly in resource-constrained settings.

Future work should focus on addressing the remaining challenges in recognizing less frequent entity types and handling informal medical descriptions. Additionally, the investigation into further performance of Word2Vec and ELECTRA models deserves more attention, as our study prioritized optimization of BERT-based models. Furthermore, evaluating these models on a larger and more evenly-distributed dataset would provide more robust insights. The current CADEC corpus, while valuable, has some inherent imbalances in entity distribution that may affect model performance. A more balanced dataset with equal representation across different entity types would help validate our findings and potentially improve recognition of less frequent entities.

## 8 Contribution

- Phillip SHEN
  - Data preprocessing
  - BERTs and ELECTRAs model implementation
  - Result visualization and table making
  - Report writing
  - Group discussion
- Xiangyu LI
  - Word2Vec implementation
  - Group discussion

## References

[1] 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. arXiv:1904.03323 [cs.CL] https://arxiv.org/abs/1904.03323

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2016. Neural Machine Translation by Jointly Learning to Align and Translate. arXiv:1409.0473 [cs.CL] https://arxiv.org/abs/1409.0473

[3] Richard Baker, Louis S Levene, Christopher Newby, and George K Freeman. 2024. Does shortage of GPs matter? A cross-sectional study of practice population life expectancy. *British Journal of General Practice* 74, 742 (2024), e283–e289. https://doi.org/10.3399/BJGP.2023.0195 arXiv:https://bjgp.org/content/74/742/e283.full.pdf

[4] K Clark. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555* (2020).

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs.CL] https://arxiv.org/abs/1810.04805

[6] Sean R Eddy. 1996. Hidden Markov models. *Current Opinion in Structural Biology* 6, 3 (1996), 361–365. https://doi.org/10.1016/S0959-440X(96)80056-X

[7] Kamal raj Kanakarajan, Bhuvana Kundumani, and Malaikannan Sankarasubbu. 2021. BioELECTRA:Pretrained Biomedical text Encoder using Discriminators. In *Proceedings of the 20th Workshop on Biomedical Language Processing*. Association for Computational Linguistics, Online, 143–154. https://doi.org/10.18653/v1/2021.bionlp-1.16

[8] Sarvnaz Karimi, Alejandro Metke-Jimenez, Madonna Kemp, and Chen Wang. 2015. Cadec: A corpus of adverse drug event annotations. *Journal of Biomedical Informatics* 55 (2015), 73–81. https://doi.org/10.1016/j.jbi.2015.03.010

[9] Navjeet Kaur, Ashish Saha, Makul Swami, Muskan Singh, and Ravi Dalal. 2024. Bert-Ner: A Transformer-Based Approach For Named Entity Recognition. In *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. 1–7. https://doi.org/10.1109/ICCCNT61001.2024.10724703

[10] John Lafferty, Andrew McCallum, Fernando Pereira, et al. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Icml*, Vol. 1. Williamstown, MA, 3.

[11] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* 36, 4 (09 2019), 1234–1240. https://doi.org/10.1093/bioinformatics/btz682 arXiv:https://academic.oup.com/bioinformatics/article-pdf/36/4/1234/48983216/bioinformatics_36_4_1234.pdf

[12] A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* (2017).

[13] Tao Liu Weiquan Wang, Li Sun and Tian Lai. 2022. The use of E-health during the COVID-19 pandemic: a case study in China's Hubei province. *Health Sociology Review* 31, 3 (2022), 215–231. https://doi.org/10.1080/14461242.2021.1941184 arXiv:https://doi.org/10.1080/14461242.2021.1941184 PMID: 34161186.

## A Appendix

Github link to the repository where we upload the code:

https://github.com/philihpop/NER-on-CADEC