Efficient Joint Learning for Clinical Named Entity Recognition and Relation Extraction Using Fourier Networks: A Use Case in Adverse Drug Events

Anthony Yazdani, Dimitrios Proios, Hossein Rouhizadeh, Douglas Teodoro
University of Geneva, Faculty of medicine,
Department of radiology and medical informatics, Data science for digital health
firstname.lastname@unige.ch

Abstract

Current approaches for clinical information extraction are inefficient in terms of computational costs and memory consumption, hindering their application to process large-scale electronic health records (EHRs). We propose an efficient end-to-end model, the Joint-NER-RE-Fourier (JNRF), to jointly learn the tasks of named entity recognition and relation extraction for documents of variable length. The architecture uses positional encoding and unitary batch sizes to process variable length documents and uses a weight-shared Fourier network layer for low-complexity token mixing. Finally, we reach the theoretical computational complexity lower bound for relation extraction using a selective pooling strategy and distance-aware attention weights with trainable polynomial distance functions. We evaluated the JNRF architecture using the 2018 N2C2 ADE benchmark to jointly extract medication-related entities and relations in variable-length EHR summaries. JNRF outperforms rolling window BERT with selective pooling by 0.42%, while being twice as fast to train. Compared to state-of-the-art BiLSTM-CRF architectures on the N2C2 ADE benchmark, results show that the proposed approach trains 22 times faster and reduces GPU memory consumption by 1.75 folds, with a reasonable performance tradeoff of 90%, without the use of external tools, hand-crafted rules or post-processing. Given the significant carbon footprint of deep learning models and the current energy crises, these methods could support efficient and cleaner information extraction in EHRs and other types of large-scale document databases.

1 Introduction

Adverse drug events (ADEs) are defined as any injury resulting from medication use and comprise the largest category of adverse events (Leape et al., 1991; Bates et al., 1995). Serious ADEs have been estimated to cost from $30 to $137 billion in ambulatory settings in the US (Johnson and Booman, 1996), and their costs have been doubling since then (Ernst and Grizzle, 2001). Due to safety concerns, between 21% to 27% of marketed drugs in the US have received black-box warnings or have been withdrawn by the Food and Drug Administration (FDA) within the first 16 years of marketing (Frank et al., 2014).

Clinical notes stored in electronic health record (EHRs) systems are a valuable source of information for pharmacovigilance (Boland and Tatonetti, 2015). However, only 1% of ADEs recorded in EHRs are reported to ADE registries, such as the FDA Adverse Event Reporting System (FAERS), while coded diagnoses have low sensitivity for ADEs (Nadkarni, 2010; Classen et al., 2011). Recognizing medication-related entities in clinical notes, extracting relations among them, and structuring this information can help identify ADEs in early stages of the drug marketing process, thus improving patient safety (Luo et al., 2017).

The state-of-the-art for biomedical named entity recognition (NER) and relation extraction (RE) is dominated by bidirectional LSTM (Hochreiter and Schmidhuber, 1997) or BERT (Devlin et al., 2018) architectures, combined with a CRF (Lafferty et al., 2001) layer and often hand-crafted rules (Xu et al., 2017; Christopoulou et al., 2020; Wei et al., 2020; Henry et al., 2020; Fang et al., 2021). Despite the high performance of end-to-end (E2E) NER+RE models, they have some important limitations imposed by the model complexity, e.g., quadratic in terms of entity types in the CRF layer or in terms of tokens in the dot-product attention mechanisms (Sutton et al., 2012; Shen et al., 2021), which hinders their effective application in the biomedical domain due to its large number of entities and large size of free text databases.

A particularity of NER and RE for pharmacovigilance is that efficient recall of entities and relations is of utmost importance, as we would like to avoid missing a serious ADE. Nevertheless, cur-

Proceedings of the 19th International Conference on Natural Language Processing (ICON), pages 212 - 223
December 15-18, 2022 ©2022 Association for Computational Linguistics

rent approaches tend to automatically discard long distance (or inter-passage) relations (Yao et al., 2019; Christopoulou et al., 2020). Moreover, EHR documents varies significantly in length, containing from a few hundred tokens for simpler patient records up to several thousand tokens for more complex patients (e.g., chronic diseases) (Henry et al., 2020). Due to their computational complexity, these methods cannot process EHRs in their integrity without resorting to impractical and/or inefficient techniques such as windowing strategies (Ding et al., 2020; Pappagari et al., 2019; Yang et al., 2016).

Ongoing research is predominantly performance-driven, leading to a resurgence of resource-intensive models, neglecting the carbon footprint of deep learning models in favor of often marginal improvement in effectiveness (Wei et al., 2020; Knafou et al., 2020; Copara et al., 2020; Copara Zea et al., 2020; Fang et al., 2021; Naderi et al., 2021). As a consequence of the technical constraints induced by highly complex models, these methods are currently being associated to a significant excess on carbon emissions (Gibney, 2022). The most direct impact of training and deploying a machine learning model is the emission of greenhouse gases due to the increased hardware energy consumption (Ligozat and Luccioni, 2021). Therefore, a direct way to reduce the ecological impact of training and deploying machine learning models is to reduce the training and inference time, i.e., providing the community with low memory and computational cost models.

To tackle these limitations and issues, we propose the Joint-NER-RE-Fourier (JNRF) model with a reduced algorithmic complexity for information extraction. We combine positional encoding with unitary batch size training so that the model processes automatically variable size EHRs with consistent performance. We use a Fourier network to contextualize tokens with fair time and space complexity, allowing to process long documents with low-resource hardware and avoid rolling window strategies. Finally, we reach the theoretical computational complexity lower bound for relation extraction using a selective pooling strategy and distance-aware attention weights with trainable polynomial distance functions. The main contributions of this paper are as follows:

- We propose a general, lightweight, and efficient model to jointly detect clinical entities and multiple relations, while requiring low computational power and memory, without the use of external tools or handcrafted rules. The code is available at https://github.com/ds4dh/JNRF.
- We show that this model can be applied to variable length documents, without any architectural changes. More importantly, it has robust performance independent of the document size.
- To the best of our knowledge, this is the first effort to model ADE and medication extraction at the document level. Unlike existing models in the literature, we demonstrate that our approach is able to identify inter-passage relations without the need of window/input size tuning, post-processing or any further engineering.

2 Related work

The main methods to produce E2E information extraction systems are the so called *pipeline* (Sorokin and Gurevych, 2017; Chapman et al., 2018; Christopoulou et al., 2020) and *joint modeling* (Xu et al., 2017; Wei et al., 2020; Bekoulis et al., 2018; Nguyen and Verspoor, 2019; Luan et al., 2019; Wadden et al., 2019). The pipeline method consists of training two independent modules, one for NER and one for RE. These models naturally suffer from cascading errors, as the error signal from one module is not back-propagated to the other. Joint modeling aims to overcome this shortcoming by learning a unique model on a combination of NER and RE losses. Joint modeling tends to outperform pipeline methods, consistently achieving state-of-the-art performance (Wei et al., 2020; Fang et al., 2021; Bekoulis et al., 2018; Nguyen and Verspoor, 2019; Luan et al., 2019; Wadden et al., 2019). In addition, joint modeling techniques have some major advances as they allow to train two models at the same time, saving time and computation, and minimizing engineering efforts. In both cases, the E2E approach has been dominated by LSTM-CRF architectures (Xu et al., 2017; Christopoulou et al., 2020; Wei et al., 2020; Henry et al., 2020). However, they suffer from two main limitations: *i)* the computational complexity of the CRF layer (Jeong et al., 2009); and *ii)* the auto-regressive nature of the LSTM model, which prevents full parallel training (Xu et al., 2021).

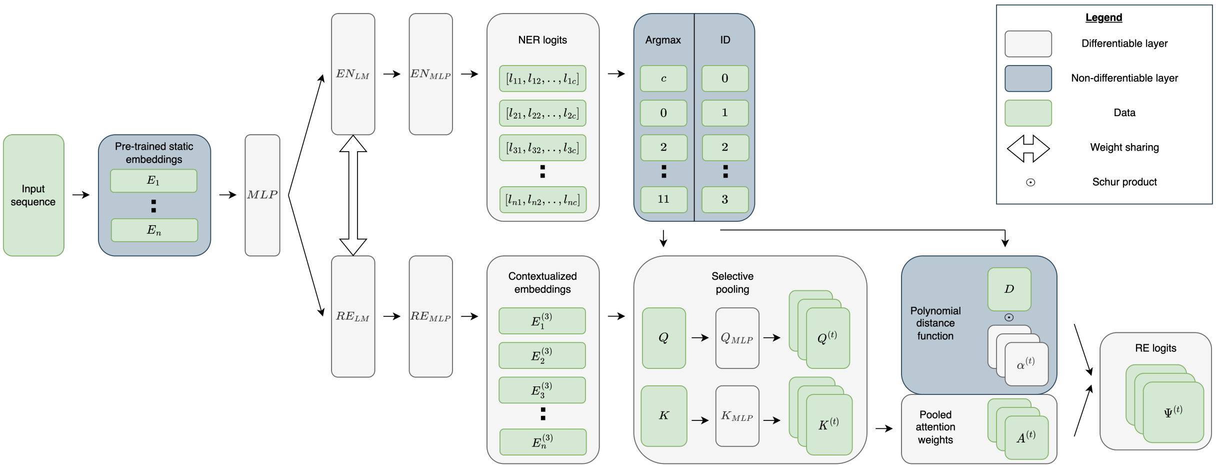

Figure 1: Computational graph for the proposed JNRF network.

2.1 Joint learning in the general domain

Bekoulis et al. (2018) proposed a joint neural model using CRFs and a multi-headed selection module allowing for multiple relation detection. The model requires the computation of scores on every pair of input tokens, which consumes $\mathcal{O}(n^2)$ time and space. To improve generalisation, their approach does not rely on external NLP tools, such as part-of-speech (POS) tagger or dependency parser. More recently, Nguyen and Verspoor (2019) proposed a joint BiLSTM-CRF architecture combined with a biaffine attention mechanism (Dozat and Manning, 2016), improving upon Bekoulis et al. (2018) in terms of time complexity. Luan et al. (2019) utilizes dynamic span graphs to learn useful information from a broader context. The graph is built by picking the most confident entity spans and linking them with confidence-weighted relation types and correlations. The model does not require preprocessing syntactic tools and significantly outperforms the previous approaches across several entity-related tasks. Lastly, DYGIE++ (Wadden et al., 2019) enumerates candidate text spans and encodes them using BERT and task-specific message updates passed over a text span graph to achieve state-of-the-art performance across entity, relation, and event extraction tasks.

2.2 Joint learning for medication-related entity and relation extraction

Most of the medication-related NER and RE studies are performed using the N2C2 ADE benchmark (Henry et al., 2020). Wei et al. (2020) proposed a system consisting of a LSTM-CRF layer for NER joint learned with a CNN-RNN layer for RE. They utilized CLAMP (Soysal et al., 2018) for the text pre-processing pipeline, including sentence boundary detection and POS labeling, and to extract a set of hand-crafted features to feed the NER module. Similarly to approaches for general corpora, Fang et al. (2021) replaced the LSTM layer by a BERT model for feature extraction, achieving 1.5 percentage point improvement in the strict F1-score metric. In their approach, a CRF layer is still used on top of a BERT model for the NER part, while a multi-head selection module (Bekoulis et al., 2018) combines the output of the BERT and CRF layers to predict relation among the detected entities.

2.3 Fourier networks

To overcome algorithmic complexity limitations in the Transformers architecture (Vaswani et al., 2017), Fourier networks (FNet) have been proposed (Lee-Thorp et al., 2021). The main innovation of FNets is that the classic Transformers attention mechanism can be mimicked using simple, non-trainable token mixing strategies. One can obtain $\mathcal{O}(n \times \log(n))$ complexity using the Cooley–Tukey Fast Fourier Transform algorithm (Cooley and Tukey, 1965) instead of the attention mechanism, which consumes $\mathcal{O}(n^2)$ with respect to the input sequence length (n). FNets achieve 92 and 97% of BERT-Base and BERT-Large (Devlin et al., 2018) accuracy on the GLUE benchmark (Wang et al., 2018), but train 70-80% faster on GPUs/TPUs. In addition to matching the accuracy of competing linear-complexity transformers (Wang et al., 2020; Jaegle et al., 2021; Wu et al., 2021; Lee-Thorp et al., 2021), the FNet is faster and memory efficient due to the unparameterized contextualization layer, i.e., it has no parameters to train for token mixing, thus requires virtually no memory usage.

3 Approach

In this section, we provide a step-by-step formal description of the proposed architecture using the forward pass representations and operations, as illustrated in Figure 1. First, we describe *i)* the vectorial token representation strategy, then *ii)* the language/contextualization layer, next *iii)* how the NER and RE task is jointly modelled, and finally *iv)* the cost functions used. Lastly, we conduct a computational complexity analysis of the proposed model.

3.1 Model formalisation

Token representation layer: We use static embeddings (BioClinicalBERT-base (Alsentzer et al., 2019) in our experiments) and freeze these parameters during training for better generalization. We also decided to use positional encoding as in Vaswani et al. (2017) so as not to fix a predefined input length.

Language model: We use FNets to perform token contextualization with fair time and space complexity. We integrate a FNet layer in our architecture as follows:

$$\begin{aligned} E^{(1)} &= \text{MLP}(E), \\ E^{(2a)} &= \text{EN}_{LM}(E^{(1)}), \\ E^{(2b)} &= \text{RE}_{LM}(E^{(1)}), \end{aligned}$$

where $E \in \mathbb{R}^{n \times d}$ is the embedding matrix, in which each row represents a token, following their order in the input sequence (i.e., the document), n the input sequence length, d the token embedding dimension, MLP is a token-wise multilayer perceptron, EN$_{LM}$ and RE$_{LM}$ are NER and RE FNets respectively. In fact, we fully share the weights between EN$_{LM}$ and RE$_{LM}$ to further reduce the number of trainable parameters. We use superscripts ($^{(1)}$, $^{(2a)}$, ...) to denote the transformed versions of the original embedding matrix.

NER and RE layers: We thus have $E^{(2)} = E^{(2a)} = E^{(2b)}$, and subsequently compute:

$$\begin{aligned} l &= \text{EN}_{MLP}(E^{(2)}), \\ E^{(3)} &= \text{RE}_{MLP}(E^{(2)}), \end{aligned}$$

where EN$_{MLP}$ and RE$_{MLP}$ are two independent token-wise MLPs. EN$_{MLP}$ maps the contextualized embeddings $E^{(2)}$ to logits $l \in \mathbb{R}^{n \times c}$ for classification, where c is the number of entity classes, and RE$_{MLP}$ maps $E^{(2)}$ to a third version of the embedding matrix $E^{(3)}$. We then compute a priori token classes

$$a_i = \text{argmax}(l_i),$$

for $i : 1 \ldots n$, and apply a selective pooling strategy, i.e., we pool candidate entities for relation extraction from $E^{(3)}$ using a_i. Some relations may never exist for a particular relation extraction task. We use L to denote the set of entities that can only be linked to those of a set H. To avoid generating impossible candidate pairs, we perform two selective pooling for these two different sets: the key $K \in \mathbb{R}^{|L| \times d}$, and the query $Q \in \mathbb{R}^{|H| \times d}$. We then produce t heads

$$\begin{aligned} K^{(j)} &= \text{K}^{(j)}_{MLP}(K), \\ Q^{(j)} &= \text{Q}^{(j)}_{MLP}(Q), \end{aligned}$$

for $j : 1 \ldots t$, where K$^{(j)}_{MLP}$ and Q$^{(j)}_{MLP}$ are token-wise MLPs, and t represent the number of relation types. We then compute the scores between the query and the key entities

$$A^{(j)} = Q^{(j)} K^{T(j)}.$$

As the RE module is distance agnostic, we incorporate a trainable polynomial distance function to modify the logits as a function of distance between tokens:

$$\Psi^{(j)} = A^{(j)} + \alpha_{j1} \times D^2 + \alpha_{j2} \times D + \alpha_{j3} \times I,$$

where $D_{\phi\psi}$ represents the number of tokens separating the ϕ^{th} and ψ^{th} pooled entities in the original input embedding matrix. The α's are learned through the minimization of the loss function and thus requires no predefined hand-crafted rules regarding short/long-distance relations.

Loss function: We use a cross-entropy loss for both NER and RE:

$$\mathcal{L}_{NER} = -\frac{1}{n} \sum_{i=1}^{n} \sum_{k=1}^{c} s(l_{i,k}) \times e_{i,k},$$

$$\mathcal{L}_{RE} = -\frac{1}{|H||L|} \sum_{h=1}^{|H|} \sum_{p=1}^{|L|} \sum_{j=1}^{t} s(\Psi^{(j)}_{h,p}) \times r_{h,p,j},$$

where $s(x_{q,z}) = \log(\exp(x_{q,z}) / \sum_b \exp(x_{q,b}))$, and e and r are the target entities and relations, respectively. Finally, we use the sum of $\mathcal{L}_{NER}$ and $\mathcal{L}_{RE}$ as the final loss function to minimize

$$\mathcal{L} = \mathcal{L}_{NER} + \mathcal{L}_{RE}.$$

3.2 Computational complexity

The complexity of the RE model depends on the number of neighbors considered for candidate pair of entities, independently of the method. If one wants to detect relations between two entities regardless of the distance, then the lower bound is $\mathcal{O}(t \times |H| \times |L|)$; or $\min(\mathcal{O}(t \times |L|)\ ,\ \mathcal{O}(t \times |H|))$ if one fixes the number of candidate neighbors. We decided not to set a maximum number of neighbors for candidate pair generation. Thus, the RE model uses $\mathcal{O}(t \times |H| \times |L|)$ through selective pooling. For a fixed RE method, the complexity of the whole model is driven by the NER component. We achieved fair complexity by using an FNet ($\mathcal{O}(n \times \log(n))$). Additionally, we used a softmax layer in place of CRF, which uses $\mathcal{O}(n \times c)$ instead of CRF's $\mathcal{O}(n \times c^2)$. This method also takes advantage of parallelization, making it a time complexity optimised method.

4 Benchmark dataset

We used the 2018 N2C2 ADE dataset [1] to evaluate our model. The data consists of 505 annotated discharge summaries from MIMIC-III (Johnson et al., 2016). The passages contains annotations for *strength*, *form*, *dosage*, *frequency*, *route*, *duration*, *reason*, and *ADE* entities, each associated with a *drug* entity. We used the official splits to train and evaluate our model, with 303 records for training and 202 for testing. Data summary statistics are presented in the Appendix A.1. *Duration* and *ADE* entities and their respective relations are not as well represented in the dataset (see Table 6). The document lengths vary widely depending on the patient's clinical history (see Table 7). There is a gap of more than 10k tokens between the smallest and largest documents (224 and 13990, respectively), which is too large to use padding efficiently. Moreover, the average document size is almost 8x larger than the typical input size of standard BERT-like implementations (4045 vs 512, respectively).

5 Experiments

We trained our models in three different data representation scenarios, where we use whole documents, sentences only, and a mixed configuration where we use both documents and sentences as training instances. Performance was then evaluated at both document and sentence levels for these different training scenarios. Our models were compared to baseline models based on MLP with selective pooling and a sliding window BioClinicalBERT-base model (WBERT) (Alsentzer et al., 2019) with selective pooling, both trained and evaluated using the whole documents.

We implemented our models using PyTorch and a single Tesla V100 GPU. We used Adam (Kingma and Ba, 2014), mini-batches of size 1 and 64 for documents and sentences, respectively. Models were trained using gradient accumulation to avoid using padding tokens. The final model was selected based on the best dev F1-score obtained during training. In the following, we present the results of our experiments using micro-lenient precision, recall, and F1-score using the challenge's official evaluation tool.

5.1 Data pre-processing

We split the provided training data into train and dev sets composed of 242 and 61 documents, respectively. We tokenize documents using BioClinicalBERT-base wordpiece tokenizer from HuggingFace (Wu et al., 2016; Wolf et al., 2019). For sentence-level modeling, we first tokenize sentences using Spacy (Honnibal and Montani, 2017) and then use aforementioned wordpiece algorithm. We encode the gold entity boundaries in the BIO scheme. The embedding matrix is initialized from BioClinicalBERT-base static embeddings. No other form a data pre-processing or external feature injection has been implemented.

5.2 End-to-end effectiveness

Table 1 shows the performance of the JNRF model in multiple settings. The best performance was obtained in the document-document setting, reaching an end-to-end F1-score of 80.49%, a precision of 91.65% and a recall of 71.76%. The JNRF outperformed WBERT with selective pooling by 0.42% in F1-score (0.09% in precision and 0.06% in recall), while reducing algorithmic complexity by one order of magnitude ($\mathcal{O}(n \times (\log(n) + c))$ *vs.* $\mathcal{O}(n \times (n + c))$). We hypothesize that using

[1] Dataset available at https://portal.dbmi.hms.harvard.edu/.

WBERT does not improve the performance due to the lack of long-range token mixing and/or an inappropriate windowing strategy. We believe that further investigation of an optimal windowing strategy could improve its performance. Moreover, we observed a significant drop in performance (37% in F1-score) when the Fnet is replaced by an MLP, demonstrating the capacity of the FNet to better attend to the correct token representations.

The JNRF model shows good performance when it is trained and evaluated with the same document representation (i.e., document-document or sentence-sentence) with similar precision in both cases and reduction in recall for the sentence-sentence setup, due to the model's limitation to detect inter-sentence relations. It is unclear though whether further data engineering could still result in equivalent performance. For the mixed training setup, the model shows stronger power to infer at the sentence level. We believe this is due to the much higher number of examples at the sentence level, which bias the model towards such representation.

Train	**Language model**	**Test**	**Precision (%)**	**Recall (%)**	**F1 (%)**
doc.	MLP	doc.	54.19	35.49	42.89
doc.	WBERT	doc.	90.66	71.70	80.07
doc.	FNet	doc.	**91.65**	**71.76**	**80.49**
		sent.	75.28	0.29	0.57
sent.	FNet	doc.	29.55	21.42	24.84
		sent.	89.50	65.80	75.84
mixed	FNet	doc.	66.99	32.83	44.07
		sent.	81.63	62.35	70.70

Table 1: Lenient micro-averaged E2E scores for different language models and document representations.

5.3 End-to-end efficiency

To compare the efficiency of our approach against architectures used in state-of-the-art approaches, we measured the time and memory used during training over 10 epochs (for the same training set) for a rolling window BERT (WBERT), a rolling window BERT-CRF (WBERT-CRF), and a BiLSTM-CRF. All window-based models used non-overlapping windows of size 512. We deliberately chose to use the minimum number of windows for these models to make them as fast as possible. Figure 2 shows the time and VRAM used by our model and state-of-the-art models. Results show that our model substantially improves upon the state-of-the-art in terms of time complexity. Forward and backward passes over the training dataset take an average of 30 seconds with our proposed architecture, while the average time for the above mentioned models is 54, 168 and 685 seconds, respectively. This increases the learning speed by a factor of 2, 6 and 22, respectively (Figure 2a). In addition, we measured an average VRAM usage of 8 GB for the JNRF architecture while the average memory usage for the above mentioned models is 4, 5 and 14 GBs, respectively. This represents a 43% GPU memory saving compared to BiLSTM-CRF (Figure 2b). WBERT and WBERT-CRF uses around 2x less memory due to the windowing strategy. This increase in efficiency is due to the fact that, differently from the quadratic complexity in terms of the number of entities c, which is generally large in the biomedical field, our model complexity has a linear dependency in terms of the number of entities, and a log-linear dependency in terms of the number of tokens (overall $\mathcal{O}(n \times (\log(n) + c))$).

5.4 Time inefficiency of windowing strategies

To demonstrate that windowing strategies are time inefficient, we measured the average forward-backward time of a rolling window JNRF (WJNRF) and its average VRAM usage (Figure 2). JNRF is 20% faster than WJNRF but WJNRF uses 26% less memory (Figure 2). While windowing strategies save VRAM, they are an inefficient solution in terms of computation time. The average document size is 4045 (see Table 7) corresponding to an average of 8 forward passes per document using standard BERT-like implementations (512 tokens maximum input size) or 28 for the longest document. So that all tokens attend to each other, we would need overlapping windows. The worst case scenario is to drag the window token-by-token, leading to 3534 ($n - WindowSize + 1$) windows on average per document.

5.5 Performance across entities, relations and document sizes

Table 2 shows the performance of our model per entity and relation types. Our model suffers from poor performance in extracting *Reason* and *ADE* entities, with an F1-score of 50.26% and 16.40%, respectively. This lower performance is also seen in other competing solutions (Henry et al., 2020). In turn, both the detection of their respective rela-

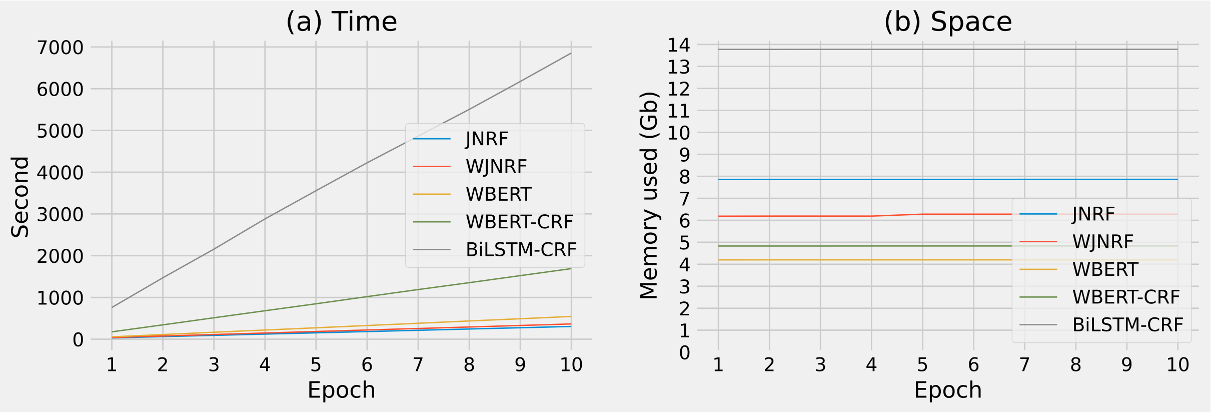

Figure 2: (a) Cumulative training time of JNRF *vs.* WJNRF *vs.* WBERT *vs.* WBERT-CRF *vs.* BiLSTM-CRF. (b) GPU memory usage of JNRF *vs.* WJNRF *vs.* WBERT *vs.* WBERT-CRF *vs.* BiLSTM-CRF. For fair comparison, all systems use the selective pooling RE module.

tions are also negatively impacted, with a final E2E F1-score of only 29.92% and 7.21%, respectively. We believe this lower performance is a result of the confusion between these entities (as they are semantically similar) and of the small number of instances in the training set. Nevertheless, further investigation is needed to better understand the issue.

Entity	Precision (%)	Recall (%)	F1 (%)
Drug	93.32	86.99	90.05
Strength	96.80	95.08	95.93
Form	96.57	92.38	94.43
Dosage	94.26	87.62	90.82
Frequency	97.63	92.37	94.93
Route	92.63	93.03	92.83
Duration	84.98	61.38	71.27
Reason	65.96	40.59	50.26
ADE	36.67	10.56	16.40
Overall	92.95	84.76	88.67
Entity + Relation	**Precision (%)**	**Recall (%)**	**F1 (%)**
Strength-Drug	95.60	88.60	91.97
Form-Drug	95.63	87.01	91.12
Dosage-Drug	94.13	79.07	85.94
Frequency-Drug	94.83	83.19	88.63
Route-Drug	90.50	83.25	86.72
Duration-Drug	76.09	41.08	53.35
Reason-Drug	54.72	20.59	29.92
ADE-Drug	30.30	4.09	7.21
Overall	90.97	72.08	80.43

Table 2: NER and E2E (NER+RE) performance of our JNRF model.

Table 3 shows the performance as a function of the number of input tokens (document length). We followed the Freedman-Diaconis method (Freedman and Diaconis, 1981) to group documents into clusters of different lengths. These results highlights the ability of our architecture to perform consistently across clinical notes of varying sizes. Without any data pre-processing (e.g., sliding window or sentence tokenization), the model can elegantly generalise to document of different sizes.

Doc. length	Doc. count	Precision (%)	Recall (%)	F1 (%)
[0, 754]	5	91.67	59.46	72.13
[754, 1508]	8	97.25	67.22	79.49
[1508, 2262]	18	89.92	66.55	76.49
[2262, 3016]	28	91.65	74.69	82.30
[3016, 3770]	43	91.72	73.46	81.58
[3770, 4524]	30	90.35	71.64	79.91
[4524, 5278]	32	90.82	72.60	80.69
[5278, 6032]	18	89.58	72.16	79.93
[6032, 6786]	10	93.88	70.99	80.85
[6786, 7540]	4	89.17	70.01	78.44
[7540, 8294]	3	88.89	67.06	76.45
[8294, 9048]	1	92.08	75.61	83.04
[9802, 10556]	1	88.73	66.55	76.06
[12064, 12818]	1	92.54	80.84	86.30

Table 3: Performance of our JNRF model across different document sizes.

5.6 Performance on long range relations

Figure 3 shows the distribution of relation types according to their sentence distance. We define the sentence distance between two related entities $E1$

and $E2$ as the number of sentences separating $E1$ from $E2$. A negative distance implies that the *drug* entity is mentioned before the related entity. Results show that although most related entities are in the same sentence, there are a non-negligible number of relations with a sentence distance different from zero. As we can see from Table 4, the JNRF model is able to automatically detect distant relations. It has superior performance detecting intra-sentence relations, i.e., better F1-score for sentence distance 0, with a yet robust performance for inter-sentence relations with negative sentence distances (between 65% and 68% F1-score). The performance decreases substantially for inter-sentence relations with positive sentence distances. This is due to the fact that *Reason* and *ADE* entities and relations are actually harder to detect (see Table 2), and they represent the vast majority of relations with a positive sentence distance, as shown in Figure 3. It is important to note that using a fixed-input size models would only detect intra-sentence relations or inter-sentence through significant engineering, which may not necessarily generalise to other corpora and domains.

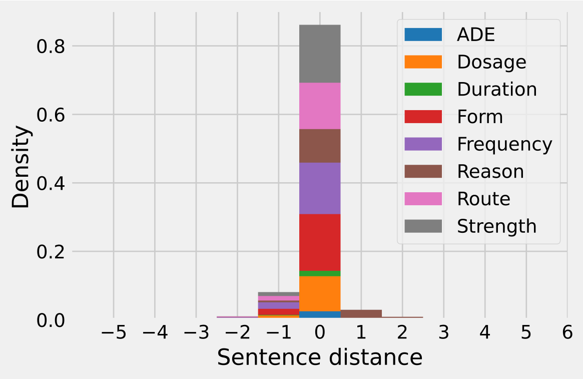

Figure 3: Probability density estimation of relation types as a function of the number of sentences separating two related entities (Sentence distance).

	Sentence distance				
	-2	**-1**	**0**	**1**	**2**
Precision (%)	75.14	83.06	92.69	22.99	0.36
Recall (%)	56.90	57.88	76.08	5.82	0.49
F1-score (%)	64.76	68.22	83.57	9.29	0.41

Table 4: Performance of our JNRF model as a function of sentence distance.

6 Comparison with SOTA in the N2C2 ADE challenge

In this section, for a reference we show our results against state-of-the-art E2E NER+RE models described in the N2C2 ADE challenge (Henry et al., 2020). Nevertheless, due to their different modelling strategy (e.g., multiple models, external tools, post-processing techniques and hand-crafted rules specifically designed for this dataset), they are not directly comparable.

UTH (Wei et al., 2020) used a joint learning model consisting of a LSTM-CRF layer for NER and a CNN-RNN layer for RE. CLAMP (Soysal et al., 2018) was employed for text pre-processing, including sentence boundary detection and POS labeling, and to create a set of hand-crafted features that fed the CRF layer. Entities without a relation were associated to the closest drug in the post-processing step.

NaCT (Christopoulou et al., 2020) used a majority voting ensemble of feature-based CRF, including ADE dictionary, and stacked BiLSTM-CRF for NER. For RE, they used an ensemble of LSTM for intra-sentence relations and a transformer network for inter-sentence relations.

BCH (Miller et al., 2019) used SVM to detect entities, and pair these detected entities for a second SVM relation classifier. They used cTAKES (Savova et al., 2010) to pre-process data and ClearTK (Bethard et al., 2014) API to extract features.

RA (Henry et al., 2020) used dictionary-based features, CRFs and logistic regression for NER. For RE, they used a tree-based boosting classifier (Chen and Guestrin, 2016).

Table 5 shows the performance of our best model as well as the results of the previously described systems. As we can see, the performance of our E2E model (80.49% F1-score) achieves 90% of the F1-score of the best performing system (99% precision and 84% recall), while significantly reducing algorithmic complexity. Moreover, it compares favorably to strong baseline methods (Chen and Guestrin, 2016) (80.49% *vs.* 80.37%), again with an order of magnitude in complexity reduction.

Name	NER complexity	Precision (%)	Recall (%)	F1 (%)
UTH	nc^2	**92.92**	**85.49**	**89.05**
NaCT	nc^2	92.64	83.18	87.66
BCH	n^3	89.63	76.40	82.49
JNRF	$n(\log(n)+c)$	91.65^a	71.76^b	80.49^c
RA	nc^2	86.89	74.75	80.37

Table 5: E2E scores of the top performing systems submitted in the N2C2 ADE track, along with our JNRF model. Standard deviations: a=0.47, b=0.53, c=0.33.

7 Conclusion

In this paper, we proposed an end-to-end, generalizable, lightweight, and efficient model to jointly detect entities and multiple relations at the intra- and inter-passage levels. We combined a Fourier network with a pooled attention layer to significantly reduce time and space complexity, thus providing the community with a low carbon footprint solution for end-to-end relation extraction. We demonstrated that our model outperformed the sliding window BERT with selective pooling by 0.42% in F1-score, while being 2 times faster to train. Furthermore, we showed that our model trains 22 times faster and consumes 1.75 times less GPU memory than state-of-the-art BiLSTM-CRF architectures, with a reasonable performance tradeoff of 90% on the N2C2 ADE benchmark, without using external tools or hand-crafted rules. Furthermore, we showed that this approach achieves consistent performance regardless of the length of the input sequence, eliminating the need for sliding window techniques and easing the overall data processing pipeline and engineering effort.

References

Emily Alsentzer, John R Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical bert embeddings. *arXiv preprint arXiv:1904.03323*.

David W Bates, David J Cullen, Nan Laird, Laura A Petersen, Stephen D Small, Deborah Servi, Glenn Laffel, Bobbie J Sweitzer, Brian F Shea, Robert Hallisey, et al. 1995. Incidence of adverse drug events and potential adverse drug events: implications for prevention. *Jama*, 274(1):29–34.

Giannis Bekoulis, Johannes Deleu, Thomas Demeester, and Chris Develder. 2018. Joint entity recognition and relation extraction as a multi-head selection problem. *Expert Systems with Applications*, 114:34–45.

Steven Bethard, Philip Ogren, and Lee Becker. 2014. ClearTK 2.0: Design patterns for machine learning in UIMA. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3289–3293, Reykjavik, Iceland. European Language Resources Association (ELRA).

Mary Regina Boland and Nicholas P Tatonetti. 2015. Are all vaccines created equal? using electronic health records to discover vaccines associated with clinician-coded adverse events. *AMIA Summits on Translational Science Proceedings*, 2015:196.

Alec B Chapman, Kelly S Peterson, Patrick R Alba, Scott L DuVall, and Olga V Patterson. 2018. Hybrid system for adverse drug event detection. In *International Workshop on Medication and Adverse Drug Event Detection*, pages 16–24. PMLR.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.

Fenia Christopoulou, Thy Thy Tran, Sunil Kumar Sahu, Makoto Miwa, and Sophia Ananiadou. 2020. Adverse drug events and medication relation extraction in electronic health records with ensemble deep learning methods. *Journal of the American Medical Informatics Association*, 27(1):39–46.

David C Classen, Roger Resar, Frances Griffin, Frank Federico, Terri Frankel, Nancy Kimmel, John C Whittington, Allan Frankel, Andrew Seger, and Brent C James. 2011. 'global trigger tool'shows that adverse events in hospitals may be ten times greater than previously measured. *Health affairs*, 30(4):581–589.

James W. Cooley and John W. Tukey. 1965. An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, 19(90):297–301.

Jenny Copara, Julien Knafou, Nona Naderi, Claudia Moro, Patrick Ruch, and Douglas Teodoro. 2020. Contextualized french language models for biomedical named entity recognition. In *6e conférence conjointe Journées d'Études sur la Parole (JEP, 33e édition), Traitement Automatique des Langues Naturelles (TALN, 27e édition), Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RÉCITAL, 22e édition). Atelier DÉfi Fouille de Textes*, pages 36–48. ATALA; AFCP.

Jenny Linet Copara Zea, Nona Naderi, Julien David Marc Knafou, Patrick Ruch, and Douglas Teodoro. 2020. Named entity recognition in chemical patents using ensemble of contextual language models. In *Proceedings of CLEF (Conference and Labs of the Evaluation Forum) 2020 Working Notes*.

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

Ming Ding, Chang Zhou, Hongxia Yang, and Jie Tang. 2020. Cogltx: Applying bert to long texts. In *Advances in Neural Information Processing Systems*, volume 33, pages 12792–12804. Curran Associates, Inc.

Timothy Dozat and Christopher D Manning. 2016. Deep biaffine attention for neural dependency parsing. *arXiv preprint arXiv:1611.01734*.

Frank R Ernst and Amy J Grizzle. 2001. Drug-related morbidity and mortality: updating the cost-of-illness model. *Journal of the American Pharmaceutical Association (1996)*, 41(2):192–199.

Xintao Fang, Yuting Song, and Akira Maeda. 2021. Joint extraction of clinical entities and relations using multi-head selection method. In *2021 International Conference on Asian Language Processing (IALP)*, pages 99–104. IEEE.

Cassie Frank, David U Himmelstein, Steffie Woolhandler, David H Bor, Sidney M Wolfe, Orlaith Heymann, Leah Zallman, and Karen E Lasser. 2014. Era of faster fda drug approval has also seen increased black-box warnings and market withdrawals. *Health affairs*, 33(8):1453–1459.

David Freedman and Persi Diaconis. 1981. On the histogram as a density estimator: L 2 theory. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 57(4):453–476.

Elizabeth Gibney. 2022. How to shrink ai's ballooning carbon footprint. *Nature*, 607(7920):648–648.

Sam Henry, Kevin Buchan, Michele Filannino, Amber Stubbs, and Ozlem Uzuner. 2020. 2018 n2c2 shared task on adverse drug events and medication extraction in electronic health records. *Journal of the American Medical Informatics Association*, 27(1):3–12.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Andrew Jaegle, Felix Gimeno, Andrew Brock, Andrew Zisserman, Oriol Vinyals, and Joao Carreira. 2021. Perceiver: General perception with iterative attention. *arXiv preprint arXiv:2103.03206*.

Minwoo Jeong, Chin-Yew Lin, and Gary Geunbae Lee. 2009. Efficient inference of CRFs for large-scale natural language data. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 281–284, Suntec, Singapore. Association for Computational Linguistics.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9.

Jeffery Johnson and Lyle Booman. 1996. Drug-related morbidity and mortality. *Journal of Managed Care Pharmacy*, 2(1):39–47.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv:1412.6980*.

Julien David Marc Knafou, Nona Naderi, Jenny Linet Copara Zea, Douglas Teodoro, and Patrick Ruch. 2020. Bitem at wnut 2020 shared task-1: Named entity recognition over wet lab protocols using an ensemble of contextual language models. In *Proceedings of the 2020 EMNLP Workshop W-NUT: The Sixth Workshop on Noisy User-generated Text*, pages 305–313. Association for Computational Linguistics.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Lucian L Leape, Troyen A Brennan, Nan Laird, Ann G Lawthers, A Russell Localio, Benjamin A Barnes, Liesi Hebert, Joseph P Newhouse, Paul C Weiler, and Howard Hiatt. 1991. The nature of adverse events in hospitalized patients: results of the harvard medical practice study ii. *New England journal of medicine*, 324(6):377–384.

James Lee-Thorp, Joshua Ainslie, Ilya Eckstein, and Santiago Ontanon. 2021. Fnet: Mixing tokens with fourier transforms. *arXiv preprint arXiv:2105.03824*.

Anne-Laure Ligozat and Sasha Luccioni. 2021. *A Practical Guide to Quantifying Carbon Emissions for Machine Learning Researchers and Practitioners*. Ph.D. thesis, MILA; LISN.

Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. 2019. A general framework for information extraction using dynamic span graphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3036–3046, Minneapolis, Minnesota. Association for Computational Linguistics.

Yuan Luo, William K Thompson, Timothy M Herr, Zexian Zeng, Mark A Berendsen, Siddhartha R Jonnalagadda, Matthew B Carson, and Justin Starren. 2017. Natural language processing for ehr-based pharmacovigilance: a structured review. *Drug safety*, 40(11):1075–1089.

Timothy Miller, Alon Geva, and Dmitriy Dligach. 2019. Extracting adverse drug event information with minimal engineering. In *Proceedings of the 2nd Clinical*

Natural Language Processing Workshop, pages 22–27, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Nona Naderi, Julien Knafou, Jenny Copara, Patrick Ruch, and Douglas Teodoro. 2021. Ensemble of deep masked language models for effective named entity recognition in health and life science corpora. *Frontiers in research metrics and analytics*, 6.

Prakash M Nadkarni. 2010. Drug safety surveillance using de-identified emr and claims data: issues and challenges. *Journal of the American Medical Informatics Association*, 17(6):671–674.

Dat Quoc Nguyen and Karin Verspoor. 2019. End-to-end neural relation extraction using deep biaffine attention. In *European Conference on Information Retrieval*, pages 729–738. Springer.

Raghavendra Pappagari, Piotr Zelasko, Jesús Villalba, Yishay Carmiel, and Najim Dehak. 2019. Hierarchical transformers for long document classification. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 838–844. IEEE.

Guergana K Savova, James J Masanz, Philip V Ogren, Jiaping Zheng, Sunghwan Sohn, Karin C Kipper-Schuler, and Christopher G Chute. 2010. Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association*, 17(5):507–513.

Zhuoran Shen, Mingyuan Zhang, Haiyu Zhao, Shuai Yi, and Hongsheng Li. 2021. Efficient attention: Attention with linear complexities. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3531–3539.

Daniil Sorokin and Iryna Gurevych. 2017. Context-aware representations for knowledge base relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1784–1789, Copenhagen, Denmark. Association for Computational Linguistics.

Ergin Soysal, Jingqi Wang, Min Jiang, Yonghui Wu, Serguei Pakhomov, Hongfang Liu, and Hua Xu. 2018. Clamp–a toolkit for efficiently building customized clinical natural language processing pipelines. *Journal of the American Medical Informatics Association*, 25(3):331–336.

Charles Sutton, Andrew McCallum, et al. 2012. An introduction to conditional random fields. *Foundations and Trends® in Machine Learning*, 4(4):267–373.

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.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5784–5789, Hong Kong, China. Association for Computational Linguistics.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. 2020. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*.

Qiang Wei, Zongcheng Ji, Zhiheng Li, Jingcheng Du, Jingqi Wang, Jun Xu, Yang Xiang, Firat Tiryaki, Stephen Wu, Yaoyun Zhang, et al. 2020. A study of deep learning approaches for medication and adverse drug event extraction from clinical text. *Journal of the American Medical Informatics Association*, 27(1):13–21.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Chuhan Wu, Fangzhao Wu, Tao Qi, Yongfeng Huang, and Xing Xie. 2021. Fastformer: Additive attention can be all you need. *arXiv preprint arXiv:2108.09084*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv:1609.08144*.

Hongfei Xu, Qiuhui Liu, Josef van Genabith, Deyi Xiong, and Meng Zhang. 2021. Multi-head highly parallelized lstm decoder for neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 273–282.

Jun Xu, Hee-Jin Lee, Zongcheng Ji, Jingqi Wang, Qiang Wei, and Hua Xu. 2017. Uth_ccb system for adverse drug reaction extraction from drug labels at tac-adr 2017. In *TAC*.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North*

American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1480–1489, San Diego, California. Association for Computational Linguistics.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. Docred: A large-scale document-level relation extraction dataset. *arXiv preprint arXiv:1906.06127*.

A Appendix

A.1 N2C2 dataset summary statistics

Entity type	**Full (%)**	**Training**	**Test**
Drug	26.8k (32)	16.2k	10.6k
Strength	10.9k (13)	6.7k	4.2k
Form	11.0k (13)	6.7k	4.4k
Dosage	6.9k (8)	4.2k	2.7k
Frequency	10.3k (12)	6.3k	4.0k
Route	9.0k (11)	5.5k	3.5k
Duration	1.0k (1)	0.6k	0.4k
Reason	6.4k (8)	3.9k	2.5k
ADE	16k (2)	1.0k	0.6k
Total	83.8k (100)	51.0k	32.9k
Relation type	**Full (%)**	**Training**	**Test**
Strength-Drug	10.9k (18)	6.7k	4.2k
Form-Drug	11.0k (19)	6.7k	4.4k
Dosage-Drug	6.9k (11)	4.2k	2.7k
Frequency-Drug	10.3k (17)	6.3k	4.0k
Route-Drug	9.1k (15)	5.5k	3.5k
Duration-Drug	1.1k (2)	0.6k	0.4k
Reason-Drug	8.6 (15)	5.2k	3.4k
ADE-Drug	1.8 (3)	1.1k	0.7k
Total	59.8 (100)	36.4k	23.5k

Table 6: Entity and relation distributions.

	Train set	**Validation set**	**Test set**
Count	242	61	202
Mean	4045	3829	3933
Std	1972	1870	1790
Min	224	237	252
Max	13990	7845	12518

Table 7: Statistics of document length in terms of tokens.