Joint Entity and Relation Extraction in Medical Text*

Leo Xinyue Zhang[1,*], Angus Roberts[1] and Sebastian Zeki[2]

[1]King's College London, Strand, London, WC2R 2LS, United Kingdom
[2]St Thomas' Hospital, Westminster Bridge Road, London SE1 7EH, United Kingdon

Abstract
Named Entity Recognition and Relation Extraction are two fundamental tasks for medical information extraction. Typically, these tasks are done in a pipeline way. However, this approach ignores the interactions between these two tasks. In addition, modelling with two models takes time to train and deploy. There is research on modelling the two tasks together. However, some research only considers entities that are in some relations. Moreover, there is seldom research on joining modelling in the medical domain. In this paper, we implemented a promising generative joining modelling method on a medical dataset. We extend the modelling mechanism to incorporate non-relation entities into the output as a self-concept relation. As such, we are able to output the entire entities and relations in one step for medical extraction.

Keywords
medical extraction, named entity recognition, relation extraction, joint modelling,

1. Introduction

Named Entity Recognition (NER) and Relation Extraction (RE) are two fundamental tasks in Information Extraction (IE) from free text. NER is the process of identifying entities from free text, and categorise them if needed, while RE is the process of identifying any existing relations between the entities. Typically, these two tasks are done in a sequential manner, i.e. named entities are extracted first before passing on to relation extraction. However, there are two main drawbacks with this approach:

- This method disregards the interaction between NER and RE tasks[1]. Because the NER and RE module are two separated modules, the information cannot flow between the two tasks. These information can be helpful. Consider the following example, "London is the capital of the United Kingdom", the information for relation extraction "capital of" can help the NER task as that relation indicates that the left hand entity will be a city and the right hand entity will be a country, province or equivalent.
- Errors from the NER task will propagate to NE [1]. In the previous example, if London is wrongly identified as a person during the NER stage, this error will not be corrected during the RE stage.

One way to solve this problem is to model NER and RE as one task, i.e. one model creates a single output containing both entity and relation extractions. For example, sequence-to-sequence (seq2seq) models directly output relation triplets which includes entities and relations between entities [2], or graph models where the nodes are entities and edges are relations [3], or question answering models where a sequence of questions are asked, and the answer contains entities and relations [4]. In this research, we focus on joint modelling, specifically using seq2seq models to output relation triplets. This method has been previously used, for example, [5] used a bidirectional Long Short-Term Memory (bi-LSTM) encoder and decoder to output strings formatting by relation triplets. Model performance is, however, restricted by the amount of data one can use. To overcome the restrictions posed by limited amounts of data, [2] proposed a novel way to construct a large dataset from Wikipedia, and pre-train a BART-based model[6] on that dataset. This method achieved the best performance on the four tasks reported [2]. However, these end-to-end methods only generate relation triplets which means that entities that are not part of any relationship will not be extracted. These entities can be important in real life applications. In this work, we propose sequence formatting that incorporates non-relation entities into the relation triplets. We compare whether incorporating these non-relation entities improves the performance of relation extraction. The contributions of this work are two fold. First, it provides a method for researchers that benefits from end-to-end relation extraction models such as REBEL without the need to set up a separate entity model for non-relation entity extraction. Second, this work shines some insight into whether non-relation entities can help relation extractions.

Woodstock'22: Symposium on the irreproducible science, June 07–11, 2022, Woodstock, NY
*Corresponding author.
leo.xinyue.zhang@kcl.ac.uk (L. X. Zhang);
angus.roberts@kcl.ac.uk (A. Roberts); sebastian.zeki@gstt.nhs.uk (S. Zeki)

2. Introduction

Named Entity Recognition (NER) and Relation Extraction (RE) are two fundamental tasks in Information Extraction (IE) from free text. NER is the process of identifying entities from free text, and categorise them if needed, while RE is the process of identifying any existing relations between the entities. Typically, these two tasks are done in a sequential manner, i.e. named entities are extracted first before passing on to relation extraction. However, there are two main drawbacks with this approach:

- This method disregards the interaction between NER and RE tasks[1]. Because the NER and RE module are two separated modules, the information cannot flow between the two tasks. These information can be helpful. Consider the following example, "London is the capital of the United Kingdom", the information for relation extraction "capital of" can help the NER task as that relation indicates that the left hand entity will be a city and the right hand entity will be a country, province or equivalent.
- Errors from the NER task will propagate to NE [1]. In the previous example, if London is wrongly identified as a person during the NER stage, this error will not be corrected during the RE stage.

One way to solve this problem is to model NER and RE as one task, i.e. one model creates a single output containing both entity and relation extractions. For example, sequence-to-sequence (seq2seq) models directly output relation triplets which includes entities and relations between entities [2], or graph models where the nodes are entities and edges are relations [3], or question answering models where a sequence of questions are asked, and the answer contains entities and relations [4]. In this research, we focus on joint modelling, specifically using seq2seq models to output relation triplets. This method has been previously used, for example, [5] used a bidirectional Long Short-Term Memory (bi-LSTM) encoder and decoder to output strings formatting by relation triplets. Model performance is, however, restricted by the amount of data one can use. To overcome the restrictions posed by limited amounts of data, [2] proposed a novel way to construct a large dataset from Wikipedia, and pre-train a BART-based model[6] on that dataset. This method achieved the best performance on the four tasks reported [2]. However, these end-to-end methods only generate relation triplets which means that entities that are not part of any relationship will not be extracted. These entities can be important in real life applications. In this work, we propose sequence formatting that incorporates non-relation entities into the relation triplets. We compare whether incorporating these non-relation entities improves the performance of relation extraction. The contributions of this work are two fold. First, it provides a method for researchers that benefits from end-to-end relation extraction models such as REBEL without the need to set up a separate entity model for non-relation entity extraction. Second, this work shines some insight into whether non-relation entities can help relation extractions.

3. Related Work

3.1. Sequence to Sequence Modelling

Sequence to sequence (seq2seq) modelling is an important task in NLP, generating a target sequence given a source sentence. Unlike classification tasks, where the model generates a fixed-length output. Seq2seq tasks require a flexible length of output. Current seq2seq modelling uses encoder-decoder models. These models have two parts; an encoder model to encode the input sentence into some internal representation, and a decoder model to generate the output sentence from this representation. Early examples included two Recurrent Neural network based models as the encoder and decoder for machine translation [7] and text summarisation [8]. A recent trend has switched the focus to attention-based models after the proposal of the transformer model [9]. Attention-based models have shown generally better performance, and the ability for parallel computing. The BART model[6] made use of transformer architecture, which enabled it to do seq2seq tasks with no restrictions on output sequence length. BART model is pre-trained on denoised and corrupted corpora to gain language reconstruction ability. The BART model achieved state-of-the-art results on a number of text generation tasks by the time it was published.

3.2. Seq2seq model for relation extraction

In a recent work by [2], a pre-trained BART-based model for relation triplets sequence generation was proposed, named REBEL. The model output is in the format of concatenating all the relation triplets in the sentence with the help of special tokens. However, to pre-train a BART based model on relations extraction task requires a large amount of annotated data. The author proposed a method to generate a silver dataset from Wikipedia. Firstly, they extract all the Wikipedia abstraction, which is the section before the list of content[1]. In Wikipedia, the entities are usually in hyperlinks. The author then mapped these entities to WikiData, which is a collectively edited knowledge graph of relations between Wikipedia

[1]This is true by the time when REBEL was published. Now Wikipedia has moved the content list to left hand side. The abstract now can be defined as the piece of text preceding any section titles

entries, and then extracted relations of these entities. But the extracted relations from WikiData may not be necessarily expressed in selected text. For example, in sentence *Donald Trump visited President of Canada*, there is no relation between Donald Trump and President although this relation exists in WikiData. To alleviate this, the author adapted an established Natural Language Inference model. The NLI will assign a score which indicates how likely the text can entail the relation triplets. The sequence with a score below 75% will be filtered out.

4. Methodology

4.1. REBEL Model for relation extraction

We use the REBEL model as our base model. In essence, they formatted a triplet with special tokens. $< triplet >$ token marks the start of a relation triplet, tokens between $< triplet >$ and $< subj >$ are the head entity in the relation triplets, tokens between $< subj >$ and $< obj >$ are the tail entity in the relation triplets, and tokens after $< obj >$ are what the relation is. If a head entity appears in more than one relation, then the second tail relation just adds on to the first relation triplets. For example, in the following sentence *The patient needs to take paracetamol three times a day for a week.* The output sequence will be

$< triplet >$ *paracetamol* $< subj >$ *three times a day* $< obj >$ $medicine - frequence$ $< subj >$ *for a week* $< obj >$ $medicine - duration$.

4.2. Entity-incorporated REBEL Model

In this work, we proposed a novel way to incorporate entities into relation triplets, named E-REBEL. The idea is to treat entities as entity relations to themselves. For example, the entity paracetamol as medication would be treated as the following triplets $< triplet >$ *paracetamol* $< subj >$ *paracetamol* $< obj >$ $entity - medicine$. To put it in a sentence with other entities and relations, in the following sentence *The patient needs to take paracetamol three times a day for a week and ibuprofen*, the final relation triplets would be

$< triplet >$ *paracetamol* $< subj >$ *threetimesaday* $< obj >$ $medicine - frequency$ $< subj >$ *foraweek* $< obj >$ $medicine - duration$ $< subj >$ *paracetamol* $< obj >$ $entity - medicine$ $< triplet >$ *ibuprofen* $< subj >$ *ibuprofen* $< obj >$ $entity - medicine$.

In this way, all the entities will be included in the output sequence, and they can be easily extracted using the *entity*−prefix. In the original REBEL formatting, ibuprofen would be missed out in the output sequence because it does not exist in any relation triplets.

Table 1
n2c2 Concepts and Relations

Concepts	(n=83840)	Relations	(n = 61475)
Drug	26,803		
Strength	10,922	Strength-Drug	10,950
Form	11,006	Form-Drug	11,048
Dosage	6,900	Dosage-Drug	6,939
Frequency	10,293	Frequency-Drug	10,352
Route	8,987	Route-Drug	9,086
Duration	966	Duration-Drug	1,069
Reason	6,384	Reason-Drug	8,611
ADE	1579	ADE-Drug	1,841

E-REBEL can give us the power to extract non-relation entities and regular relations in one model. However we also ask if the non-relation incorporation could also enhance the performance of relation extraction and vice versa. We conducted a comparison between the REBEL and E-REBEL models.

5. Experiment Design

5.1. Dataset and Evaluation

The data used in this project is from the 2018 n2c2[2] shared task on adverse drug events and medication extraction in electronic health records[10]. The data includes 505 discharge summaries from the MIMIC-III (Medical Information Mart for Intensive Care-III) clinical care database[3]. The task defined 9 drug related concepts and 8 drug related relations. The list of concepts and relations, number of samples for each concept and relation can be found in Table 1. The challenge for this task is to distinguish whether two entities form a "Reason-Drug" relation or rather a "Drug-ADE" (Adverse Drug Event) relation. The dataset is not balanced in either entity types, or in relations.

The evaluation metrics used in this experiment include precision, recall and F1 score. They defined as follows:

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

Where TP is the number of true positives, FP is the number of false positives and FN is the number of false negatives. We choose F1 score over accuracy

[2]https://portal.dbmi.hms.harvard.edu/projects/n2c2-nlp/
[3]https://mimic.mit.edu/

Table 2
Relation performance of REBEL model on n2c2 data

Relations	precision	recall	F1
Drug-Reason	71.72	73.11	72.41
Drug-Form	90.73	88.48	89.59
Drug-Strength	92.98	91.65	92.31
Drug-ADE	66.56	69.00	67.76
Drug-Dosage	90.57	89.71	90.14
Drug-Frequency	91.74	88.44	90.06
Drug-Route	92.89	90.07	91.46
Drug-Duration	69.62	72.75	71.15
Micro	88.05	86.73	87.39
Macro	83.35	82.90	83.11

Table 3
Relation performance of E-REBEL model on n2c2 data

relations	precision	recall	F1
Drug-Reason	60.91	50.52	55.23
Drug-Form	87.58	88.01	87.80
Drug-Strength	82.02	81.78	81.90
Drug-ADE	47.77	43.85	45.73
Drug-Dosage	80.88	85.22	82.99
Drug-Frequency	77.72	57.40	66.03
Drug-Route	82.09	82.54	82.32
Drug-Duration	51.95	36.20	42.67
Micro	79.78	74.80	77.21
Macro	71.36	65.69	68.08

because the entity and relation types are very imbalanced in the data. We also calculated micro and macro overall F1 score for relations and for entities.

5.2. REBEL Framework

We adopted the pre-trained REBEL model as described in [2]. We fine-tuned the model on our dataset. We explored the following learning rates:

Learning rate: 1e-5, 2.5e-5, 5e-5, 7.5e-5, 1e-4

After this grid search, we found that the REBEL model with learning rate 7.5e-5, and E-REBEL model with learning rate 2.5e-5. We use maximum sequence length of 256 for REBEL with batch size 8, and maximum sequence length of 1024 for E-REBEL with batch size 2 because E-REBEL sequences are two to three times longer than the corresponding REBEL sequences. The batch size is due to the GPU memory limit give the sequence length.

6. Preliminary Results

The precision, recall and F1 score of end-to-end relation extraction of REBEL, E-REBEL models are shown in Table 2 and 3 respectively. The precision, recall and F1 score of entity extraction performance of E-REBEL are shown in Table 4. The F1 scores of end-to-end relation extractions of REBEL, E-REBEL and top five models on n2b2 leader board are shown in tabel 5. The F1 scores include micro and macro F1 scores and F1 scores of Drug-ADE and Drug-Reasons. The F1 scores of concept extractions of E-REBEL and top five models on n2b2 leader board are shown in tabel 6. These five models are not necessarily the same as end-to-end relation top five models. Micro and macro F1 scores and F1 scores of ADE and Reasons are shown. All rankings are based on micro-F1 score across all relations or across all concepts.

Table 4
Entity performance of E-REBEL model on n2c2 data

Entities	precision	recall	F1
Drug	89.84	89.70	89.77
From	86.54	69.77	77.25
Reason	81.60	82.44	82.02
Strength	93.46	92.59	93.02
ADE	74.60	79.21	76.84
Dosage	82.29	90.80	86.34
Frequency	95.76	95.96	95.86
Route	82.26	81.38	81.82
Duration	76.70	78.49	77.59
Micro	89.20	89.33	89.27
Macro	84.78	84.48	84.50

Table 5
REBEL based model compare to other models on end-to-end relations extraction. Best scores for each column are in bold. Scores that are read from graph are marked with *

	Drug-ADE	Drug-Reason	Micro	Macro
UTH	48.00*	57.00*	**89.05**	**86.37**
UFL	41.00*	59.00*	87.78	85.29
NaCT	25.15	55.05	87.66	84.23
MSC	23.00*	55.00*	86.88	83.35
VA	35.00*	48.00*	86.55	83.67
REBEL	**67.76**	**72.41**	87.39	83.11
E-REBEL	45.73	55.23	77.21	68.08

7. Analysis

From Table 2 and 5, REBEL model achieves relatively good performance on end-to-end relation extraction. Its micro and macro F1 scores are on a par with top models on the leader board. Markedly, its performance on Drug-Reason and Drug-ADE relations is far better when compared to other models. REBEL model has a relatively balanced scores on precision and recall.

E-REBEL model has a decreased performance when

Table 6
E-REBEL model compare to other models on entity extraction. Best scores for each column are in bold. Scores that are read from graph are marked with *

	ADE	Reason	Micro	Macro
Ali	58.00*	72.00*	**94.18**	**93.59**
UTH	52.00*	68.00*	93.45	92.59
UFL	46.00*	79.00*	92.87	92.26
UM	48.00*	66.00*	92.67	91.96
MSC	27.00*	58.00*	92.66	91.48
E-REBEL	**76.84**	**82.02**	89.27	84.50

compared to REBEL model. The drop is on all relations but there are some big drops from the recalls of Drug-Reason, Drug-ADE, Drug-Duration and Drug-Frequency. This means the model is more conservative on generating relation triplets. This shows that integrating entity triplets may not help with the relation triplets generations. A possible explanation is that the E-REBEL model has to generate sequences that are much longer than the REBEL model does, and within each output sequence, entity parts are generally longer than relation parts. This increases the difficulty of generating more and accurate relation triplets. Additionally, we use REBEL pre-trained model for fine-tuning which is not trained on medical specific data, and it does not have entity incorporating. Lastly, We need to test on more dataset to reach a more convincing conclusion on whether the entity incorporating decreases the relation extraction, especially including dataset that has good amount of non-relation entities.

8. Future work

This paper is a working progress. There are four aspects that I am working on.

Data The data used in this paper is limited. I plan to use more medical data to have a better understanding of E-REBEL model performance. especially for the data that includes entities that are not always in some relations.

Entity incorporating Method There are other ways to incorporate entities into output sequence. I am currently working on some possible methods and to compare the performance of these methods.

Entity incorporated Retraining In this work, we only Incorporated entities in fine tuning stage. The pre-trained REBEL model does not have entity incorporation. This impairs the performance of E-REBEL model and lead to unfair comparison between REBEL and E-REBEL models.

Medical knowledge integration REBEL model are pre-trained on Wikipedia data, which is a collection of general language information. I plan to create a medical REBEL dataset for model to gain domain knowledge.

There main real life applications coming out of this work if it succeeds is a pre-trained entity incorporated REBEL that can serve as a general framework for downstream medical entity and relation extraction tasks such as extracting information from endoscopy, pathology, radiology reports. For example, this pretrained model will be used for my PhD project which involved extracting entities and relations from pathology and endoscopy reports for Barrett's oesophagus patients.

References

[1] M. Miwa, M. Bansal, End-to-end relation extraction using lstms on sequences and tree structures, arXiv preprint arXiv:1601.00770 (2016).

[2] P.-L. H. Cabot, R. Navigli, Rebel: Relation extraction by end-to-end language generation, in: Findings of the Association for Computational Linguistics: EMNLP 2021, 2021, pp. 2370–2381.

[3] T.-J. Fu, P.-H. Li, W.-Y. Ma, Graphrel: Modeling text as relational graphs for joint entity and relation extraction, in: Proceedings of the 57th annual meeting of the association for computational linguistics, 2019, pp. 1409–1418.

[4] X. Li, F. Yin, Z. Sun, X. Li, A. Yuan, D. Chai, M. Zhou, J. Li, Entity-relation extraction as multi-turn question answering, arXiv preprint arXiv:1905.05529 (2019).

[5] T. Nayak, H. T. Ng, Effective modeling of encoder-decoder architecture for joint entity and relation extraction, in: Proceedings of the AAAI conference on artificial intelligence, volume 34, 2020, pp. 8528–8535.

[6] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, arXiv preprint arXiv:1910.13461 (2019).

[7] K. Cho, B. Van Merriënboer, D. Bahdanau, Y. Bengio, On the properties of neural machine translation: Encoder-decoder approaches, arXiv preprint arXiv:1409.1259 (2014).

[8] R. Nallapati, B. Xiang, B. Zhou, Sequence-to-sequence rnns for text summarization (2016).

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[10] S. Henry, K. Buchan, M. Filannino, A. Stubbs, O. Uzuner, 2018 n2c2 shared task on adverse

drug events and medication extraction in electronic health records, Journal of the American Medical Informatics Association 27 (2020) 3–12.