Natural Language Processing (NLP) for characterization of computerized tomography (CT) and X-Ray text reports

Gaoge Zhao
Bernard Hernandez
Yuting Xing
Pantelis Georgiou

2021.9

Abstract

Named entity recognition (NER) systems are commonly built using supervised methods which requires corpora manually annotated with named entities. However, manual annotation is very expensive and laborious. In this paper, a novel method is proposed for training clinical NER systems that does not require any manual annotation. It only needs a text corpus, a semantic type database like UMLS that can give entities semantic types, and a pre-trained BERT representation on biomedical data. Using these resources, with a few analyses, annotations are automatically obtained to train the BERT token classifier for NER tasks.

Keywords natural language processing, named entity recognition, UMLS, BioBERT

Acknowledgment

First, I want to show my special gratitude to my supervisor Dr. Bernard A Hernandez Perez. Your helpful advice and guidance pushed me further on design methodology and optimization, which formulated current work. I am grateful for our thorough analysis on project details, and it helped me understand many interesting findings better.

I also would like to thank Yuting Xing for her advice and reminding me to keep up with the deadline, without which I couldn't make it fast and efficiently.

Then I'd like to thank Xiulian Ceng, whose encouragement and inspiration had guided me all the time. And Yohana Cao, with whom I shared happiness and worries.

Finally, I would like to thank my parents for their continuous support during the year. Their encouragement and company through video phones made me optimistic and determined in front of stress and challenges.

Table of Contents

1 Introduction

1.1 Background

Infectious diseases are a widespread problem in UK hospitals, with around one third of all patients on normal wards, and two thirds of patients in intensive care being on antimicrobials at any time. Given the large number of patients and the high critical rate, radiographic checks are commonly conducted around their chests to diagnose diseases and determine the severity. To record and summarize findings of these checks, Electronic Medical Records are kept with films and written reports describing these findings. Taking into account the fact that patients' conditions might shift severely, one patient will probably have a dozen radiographic films and reports during a course of treatment. And a great number of patients will have a great number of films and documents, making it hard for clinicians to track and organize.

The situation was even worse during the COVID-19 pandemic. As ICUs and clinics got filled with patients, clinicians found themselves buried with CT and X-ray films alongside reports characterizing them. If they were given an automatic and efficient tool to categorize and organize those data, not only would their burdens be greatly relieved, but also their productivity could be improved, so that they could hospitalize and monitor more patients and save more lives.

To confront the challenges brought by COVID-19, as well as other infectious diseases, a demand for a more advanced clinical information manage system arose. Enhanced, Personalized, and Integrated Care for Infection Management at the Point of Care (EPiC IMPOC) is a clinical decision support system developed to improve the management of infectious diseases by facilitating data collection, infection diagnostics, and antimicrobial therapy advice.

Named Entity Recognition (NER) is one of the key technologies of natural language processing. Its main function is to identify specific types of entities from massive texts, such as organs, body parts, and disorders in radiology reports. Named entity recognition is an indispensable part of a variety of natural language processing technologies such as information extraction, information retrieval, knowledge graphs, machine translation, and question answering systems. It plays an essential role in constructing a modern digital medical information management system.

The named entity recognition of electronic medical records is to distinguish medical entities from normal concepts and classify these medical entities into different categories. Because there are many medical terms and special symbols in electronic medical records, the description in medical records is different from that in general texts, which makes the natural language processing technology used in the open field perform poorly on electronic medical records.

To tackle this issue, one common way is to build a named entity recognition system specifically for electronic medical records. Many efforts had been made by other researchers for this purpose. However, most of their solutions were developed to identify one or two types of named entities or for subjects other than infectious diseases. Given the data available and the personalized demand for recognizing multiple types of entities at the same time, we decide to build radiographic reports named entity recognition system on our own.

For this project, our goal is to develop a named entity recognition system to characterize the text reports produced by radiologists for computerized tomography (CT) and X-Ray reports. The expected output is a set of tags describing the CT/X-Ray reports that will be presented to the clinicians at the point of care to summarize all the radiology findings. Thus, this system can then be integrated into and play a part in the EPiC IMPOC clinical decision support system.

1.2 Scope and objectives

The objectives of this work include:

1) Extract radiograph reports out of the csv file provided.
2) Divide the reports extracted into training and test sets.
3) Conduct a statistic on the training set to find out categories of valuable entities.
4) Conclude and merge entity types found to develop a labeling system that is efficient and systematic enough for this project.
5) Use this labeling system to annotate the whole dataset.

6) Train an efficient named entity recognition model with the training set.

7) Based on the labels acquired, build a negation detection mechanism to alert clinicians of existing diseases and symptoms in the report.

8) Evaluate this model with the test set.

1.3 Achievements

After more than two months' development, the majority of our initial goals were met, which include:

1) Reports successfully extracted and saved as multiple txt files for different purposes.

2) Six important classes of entities, which are anatomy concept, disorder, qualitative concept, spatial concept, medical device, and diagnostic procedure, were identified and established as a novel labelling system for this project.

3) All data were labeled with an automatic annotation tool called QuickUMLS.

4) BERT was chosen as the model to use, and a BERT representation specifically developed for processing biomedical information called BioBERT was selected and fine-tuned for this task.

5) Assessing the BioBERT model trained, we obtained a weighted average recall of 0.942, a weighted average precision of 0.6614, and a weighted average F1 score of 0.7741.

6) A simple yet efficient algorithm was invented, which will give out the warning if there exists a disorder and is **NOT** a negation in the same sentence.

1.4 Overview of dissertation

The dissertation will cover the following topics: literature review, materials and methods, results and discussion, and conclusion.

In the literature review, we will describe several important works previously done in the field of biomedical named entity recognition, some of which will be used for ours. Those efforts can be divided into two categories, rule-based and statistic-based.

Then we will look into the problem itself in the section of materials and methods. Starting from a column in the data provided by NHS, we will explain how we extracted reports, preprocessed, used QuickUMLS to generalize features and make annotation, trained a BioBERT token classifier to do named entity recognition, and used the named entities recognized to do negation detection.

In the next part we will present the results of our BioBERT token classifier and negation detector. Also, we will present some erroneous cases and discuss why they happened.

In the last section, we will summarize the completion of our project and propose some directions in which we can make it better.

2 Literature Review

The classification of the named entity recognition methods is not unique, and different researchers have different classification perspectives, thus forming a variety of classification methods. Today, the classification method recognized by most researchers is based on the stage of technological development. In this section, we divide named entity recognition tasks into three categories according to technological development stages: dictionary-based and rule-based, statistical machine learning, and deep learning.

2.1 Dictionary-based and Rule-based NER

The method based on rules and dictionaries is the first method used for named entity recognition. This method is used by researchers to construct corresponding special dictionaries or rule templates based on text features manually [1-3] and then perform entity recognition through pattern matching. Among them, the rules include word position, keywords, demonstrative words, central words, localizers, punctuation marks, statistical information, etc. A dictionary is composed of two parts: an external dictionary and a characteristic word dictionary. The external dictionary refers to a well-known common-sense dictionary which is made up of words used in general domains and daily life, such as “the”, “next”, and “place”. The feature word dictionary refers to a collection of feature words extracted after text parsing. In the clinical domain, it contains words that appear often in medical texts but seldom in general fields, such as “pleural”, “consolidation”, and “oedema”. After manually formulating rules and dictionaries, the matching method is used to realize named entity recognition on the text.

This method mainly relies on terminology dictionaries and experts’ knowledge. In 1995, MedLEE[4], a natural language processing system, came out. Its main task was to extract, structure, and encode clinical information from patients’ reports to integrate it with the clinical information system. The system was also used by Sevenster M. et al. in 2012 to extract the relationship between human organs and clinical findings from radiology reports and obtained an accuracy of 82.32%~91.37%[5]. In the past 20 years, one of the important reasons why the MedLEE system can still achieve good results is the support of a large dictionary of medical terms. The Unified Medical Language System[6] (UMLS) developed by the National Library of Medicine since 1986 and the Medical Entities Dictionary[7] (MED) created on this basis have laid a foundation for rule-based medical entity recognition.

The named entity recognition method using dictionaries and rules benefits from the wisdom of experts in the professional field and can compile models suitable for the rules of the industry according to the text characteristics of different industries. At the same time, it is also limited by this characteristic, and the generalization of such methods is extremely weak: if a model with a good effect in a certain field is used to recognize text in another field, the effect will be greatly reduced. With the progress of society, the boundaries between various industries are becoming more and more blurred, and it is impossible for methods based on dictionaries and rules to adapt to rule models in all fields. In addition, before establishing dictionaries and rules, researchers must fully learn grammar and language structure, which requires much time for researchers. It is also necessary to conduct multiple evaluations and repeated revisions in the formulation to be perfected. Especially after the rules are determined, the sentence structure needs to be parsed through the rules to extract entities, and the whole process is tedious and lengthy.

The limitations of the method based on dictionaries and rules are obvious. It not only consumes a lot of manpower and time but cannot be extended to entities or data sets in other fields, and cannot adapt to changes in data between fields. Therefore, methods based on dictionaries and rules soon faded out of the task of named entity recognition, ushering in the era of machine learning.

2.2 Statistical Machine Learning NER

With the development of natural language processing and the expansion of text data sets in various fields, the research direction of named entity recognition technology has gradually turned to statistical machine learning methods. The principle of this method is to count the relevant parameters and features from the text data set and then establish a recognition model. This type of method is a relatively common and traditional model in the development of artificial intelligence in recent years.

In methods based on statistical machine learning, named entity recognition is regarded as a sequence labeling problem. Different from the classification problem, the current prediction label in the sequence labeling task is affected by both the current input feature and the previously predicted label, which means that there is a strong interdependence between the entity label sequences that need to be predicted. The traditional machine learning methods used mainly include the hidden Markov model, support vector machine,

conditional random field, maximum direct Markov model, maximum direct, naive Bayes model, etc.

Statistical methods had been intensively looked into in the past decades. Li Y. and Gorman S. L.[8] used 9679 English clinical reports to establish a hidden Markov model and identified the order of appearance of different modules (such as chief complaint, allergy, family history, and surgical history) in the report. Although statistics-based methods reduce the requirements for linguistic and medical knowledge to a certain extent, researchers still need to select features by themselves. The quality of feature selection will directly affect the recognition effect. In recent years, the development of deep learning has provided a method for automatically extracting word features, which helps to further reduce the feature dependence in the use of the method and improve the stability of the recognition results.

2.3 Deep Learning NER

Deep learning is a branch of the machine learning field and was first proposed by Hinton et al.[9]. The motivation lies in simulating and establishing the neural network of the human brain for learning and imitating the mechanism of the human brain to analyze data such as text and images, which belongs to unsupervised learning. This technology can get rid of domain knowledge and complex feature engineering, improve performance by adjusting parameters, and achieve excellent results in various NLP tasks.

Collobert R. et al.[10] first systematically applied deep learning to information extraction in 2011. They achieved the automatic vectorized representation of words based on neural networks, which greatly enhanced the portability of the model. After the publication of this article, deep learning methods have gradually become the focus of research in the field of named entity recognition. Socher R. et al.[11] proposed the MV-RNN model, which uses a recurrent neural network (RNN) to extract the meaning of words and uses a matrix of word vectors to extract constraint features between words. It has achieved better results than the RNN model. Lample G. et al.[12] also proposed the BiLSTM-CRF model taking into account the influence between words. The output of the Bidirectional Long Short-Term Memory (BiLSTM) model is used as the input of CRF, through which the inter-tag state transition matrix is introduced. Trained with semantic score and contextual information score of words, the BiLSTM-CRF model performs better than the BiLSTM or CRF model alone in the open corpus tests in English, German, Dutch, and Spanish.

However, there is a problem with the above methods: these methods cannot represent the polysemy of a word, as they mainly focus on extracting features of words, characters or between words, but ignore the context or semantics of the word context. So, they can only extract static word vectors that does not contain contextual information, which leads to a decline in their performance. To deal with this problem, Jacob Devlin et al.[13] from Google proposed BERT (Bidirectional Encoder Representation from Transformers). As an advanced pre-training word vector model, BERT further enhances the generalization ability of the word vectorization model. And it can fully describe relationship features at character, word, sentence, and even inter-sentence level. These make it represent the syntactic and semantic information in different contexts better. Jana Straková et al.[14] applied pre-trained BERT on CoNLL-2002 Dutch, Spanish and CoNLL-2003 English and obtained quite good results.

For our project, we first conducted feature selection and annotation with UMLS. Then, with the annotated data, we trained a BioBERT token classifier to reduce the size and improve performance.

3 Materials and Methods

Our method requires three resources – a raw corpus, a semantic type data base like UMLS, and a pretrained BioBERT embedding. And the overview of the training evaluation process is shown in Figure 1.

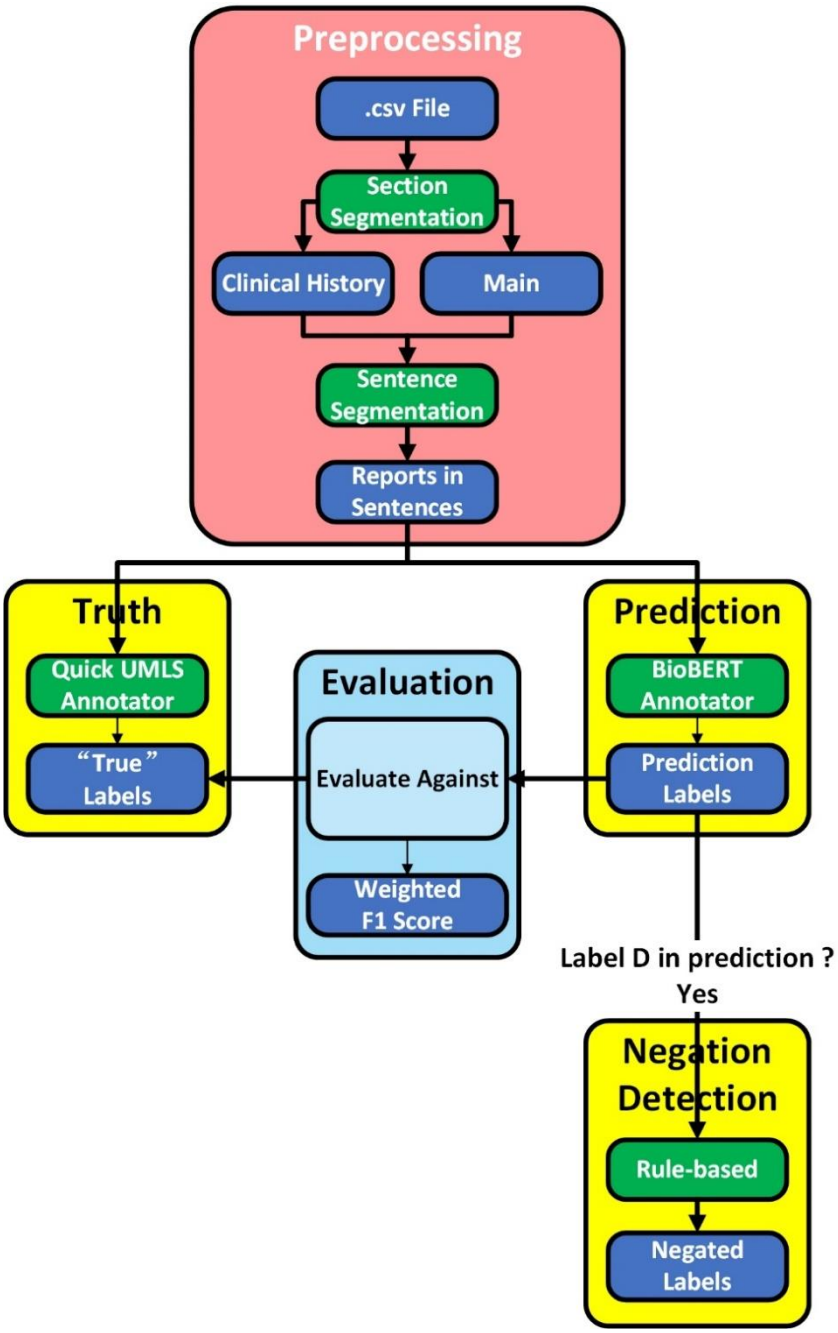

Figure 1. Training and evaluation overview.

3.1 Data overview

The data provided was in the form of a .csv file. While there are many columns addressing several categories of information that might be useful, only the one titled "diagnostic_outcome_report" is needed, for it is the place where the content of reports is held. Extracting all records in this column with the pandas library, and we get free text reports. One of the extracted reports is given in Figure 2.

Examination Date: 30/12/2019 23:27

========== REPORT TEXT START ==========

RYJ11741738 30/12/2019 XR Chest:

Clinical History

Sudden onset chest pain /cause. Please rule out pneumothorax

Report

PA erect chest radiograph.

No previous images for comparison.

The heart is not enlarged.

The lungs and pleural spaces are clear. Specifically, no focal consolidation, collapse or effusion. No pneumothorax.

No acute bony abnormality identified.

Dictated by XXXXX
Authorised by XXXX
========== REPORT TEXT END ==========

Figure 2. Report example.

3.2 Preprocessing

We then observed that while the report contains many information such as the reference number, the author and the examiner, we only need two sections of it—the Clinical History and the Report (which will be referred as "Main" to avoid conflicts). Thanks to the rigorous style of the clinical reports, section segmentation was easily done with Python's substring and slice function. Then for each section, sentence tokenization, punctuation removal and lower-casing were carried out with the sentence tokenizer from the NLTK library, regular expression, and Python's `lower` function. An intuitive view of the preprocessed data is shown in Figure 3. A total of 10575 reports were found. To make it easy for further training and evaluation, we flattened the data array and saved all preprocessed

sentences in text files. After the flattening, 64856 lines in the whole dataset were recorded.

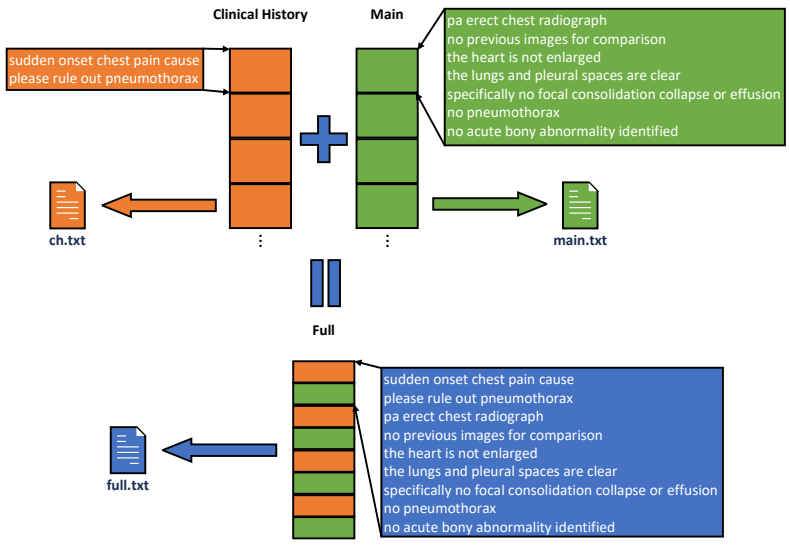

Figure 3. Preprocessed and saved data.

3.3 Data annotation

3.3.1 Unified Medical Language System

Since the data provided are unannotated, it's necessary to get them labeled first. The UMLS, or Unified Medical Language System, is a set of files and software that brings together many health and biomedical vocabularies and standards to enable interoperability between computer systems. It integrates and distributes key terminology, classification and coding standards, and associated resources to promote the creation of more effective and interoperable biomedical information systems and services, including electronic health records. For this annotation task, we would use UMLS Metathesaurus, as it records medical terms in strings along with semantic types systematically developed.

UMLS Metathesaurus is an integration of 218 source vocabularies (as of 2021AA release) across 25 different languages, including over thirteen million distinct concept names. Taking into account the fact that a term might represent multiple concepts, as well as one concept could be expressed in different terms, records in UMLS Metathesaurus are arranged in an elegant yet efficient manner, as displayed in Table 1.

The primary key of a record is the concept with a unique ID, CUI. Each unique concept name or string in each language in the Metathesaurus has at least one unique and permanent string identifier (SUI). Any variation in character set, upper-lower case, or punctuation is a separate string, with a separate SUI. If the same string, e.g., Cold, as shown in the table, has more than one meaning, the string identifier will be linked to more than one concept identifier (CUI). For English language entries in the Metathesaurus only, an English "term" is the group of all strings that are lexical variants of each other, and every string is linked to all of its lexical variants or minor variations by means of a common term identifier (LUI). At last, a concept has at least TUI, indicating its semantic type.

Concepts (CUIs)	Terms (LUIs)	Strings (SUIs)	Semantic Types (TUIs)
C0009264	L0215040 cold temperature	S7669511 Cold Temperature	T070
Cold Temperature	L0009264 cold	S0026353 Cold	Natural Phenomenon or Process
C0009443	L0009443 cold common	S0026747 Common Cold	T047
Common Cold	L0009264 cold	S0026353 Cold	Disease or Syndrome
C0024117	L0498186 airway chronic disease obstructive	S0837575 Chronic Obstructive Airway Disease	T047
Chronic Obstructive Airway Disease	L0008703 chronic disease lung obstructive	S0837576 Chronic Obstructive Lung Disease	Disease or Syndrome
	L0009264 cold	S0474508 COLD	

Table 1. UMLS Metathesaurus representation.

Unified Medical Language System has a consistent categorization of all concepts represented in the UMLS Metathesaurus called the Semantic Network, which also provides a set of useful and important

relationships, or Semantic Relations, that exist between Semantic Types. Containing 127 semantic types and 54 relationships, the Semantic Network serves as an authority for the semantic types that are assigned to concepts in the Metathesaurus. It defines these types, both with textual descriptions and by means of the information inherent in its hierarchies. An example with simplification for an easier understanding of such hierarchies using the concept "Common Cold" is given in Figure 4.

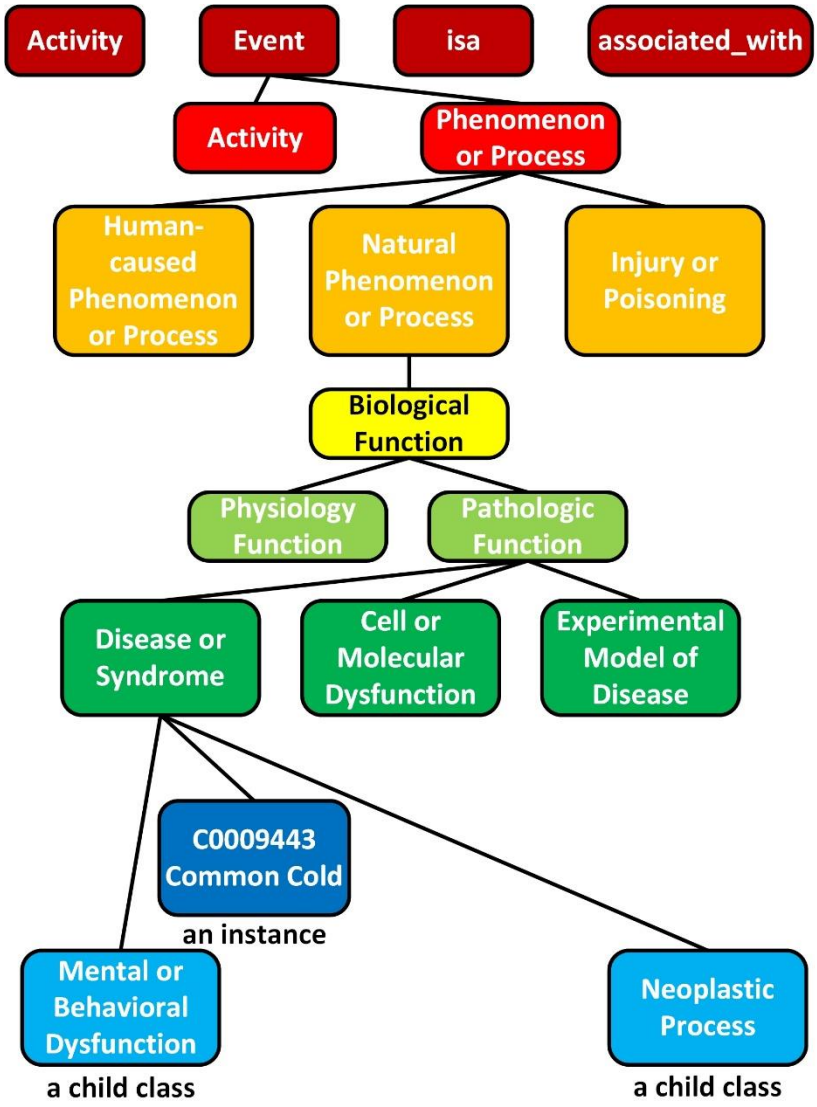

Figure 4. The hierarchical structure of the Semantic Network.

One can see from the structure of the Semantic Network that while there exist many classes of TUIs, many of them belong to broader

classes, suggesting the possibility of merging child classes into major classes for simplicity and efficiency.

3.3.2 QuickUMLS

To carry out named entity recognition with UMLS, a tool called QuickUMLS[15] was utilized. QuickUMLS is a tool for fast, unsupervised biomedical concept extraction from medical text.

The problem of extracting concepts from unstructured documents, according to Luca Soldaini and Nazli Goharian, can be formally defined as follows: given a target string x, a threshold α, a dictionary $\mathcal{S}$, and a similarity function $strsim$, we wish to find the subset $\mathcal{Y}_{x,\alpha} \subseteq \mathcal{S}$ such that

$$\mathcal{Y}_{x,\alpha} = \{y \in \mathcal{S} \mid strsim(x, y) \geq \alpha\} \tag{1}$$

For the similarity function, the authors chose Jaccard similarity:

$$Jaccard(x, y) = |x \cup y| \; / \; |x \cap y| \tag{2}$$

A naïve algorithm to compute the similarity of each string in $\mathcal{S}$ to the target string x has a complexity $O\ |\mathcal{S}|$, which is unaffordable for such a heavy database like UMLS. To solve this issue, the authors then took advantage of an algorithm called CPMerge[16] by Okazaki and Tsujii.

Instead of computing the similarity with every string in the vocabulary, CPMerge obtains $\mathcal{Y}_{x,\alpha}$ by representing strings as a set of features, and then finding strings in $\mathcal{S}$ that share more than τ features in common with x. CPMerge represents the dictionary $\mathcal{S}$ as an inverted index that associates each feature with the strings covering it. In this way, the problem of approximate dictionary matching in converted to finding a solution to the $\tau - overlap\ join$ problem[17] on the posting lists (lists of strings associated with each feature[18]) of the features of target string x.

Given a document d of length n, a similarity threshold α, and a window size w , QuickUMLS efficiently generates, for each token $d_i \in d$, all possible sequences of tokens $d_{ij} = \{d_i, \cdots, d_j\}, j \in \{i, \cdots, i + w - 1\}^4$. Then, a set of heuristics is used to determine whether d_{ij} is a valid sequence of tokens; If it is, CPMerge is used to identify strings in $\mathcal{S}$ that are similar to d_{ij}. Once the subset of all possible matching strings $\mathcal{Z}_{d,\alpha} = \cup_{d_{ij}} \left(\mathcal{Y}_{d_{ij},\alpha}\right)$ is determined, QuickUMLS selects the most appropriate subset $\mathcal{Z}'_{d,\alpha}$ of strings so that there is no overlap between the set of extracted concepts.

After fully configurated, when given a string input, it will automatically search for terms in it and retrieve their offsets in the sentence, ngrams, terms, CUIs, similarity, and most importantly, their TUIs, which are semantic types.

3.3.3 Feature selection

While QuickUMLS can retrieve many semantic types from the dataset, only a few of them are essential. To determine which semantic types to extract, a statistic on all semantic types in the dataset was carried out first, whose result is shown in Figure 5.

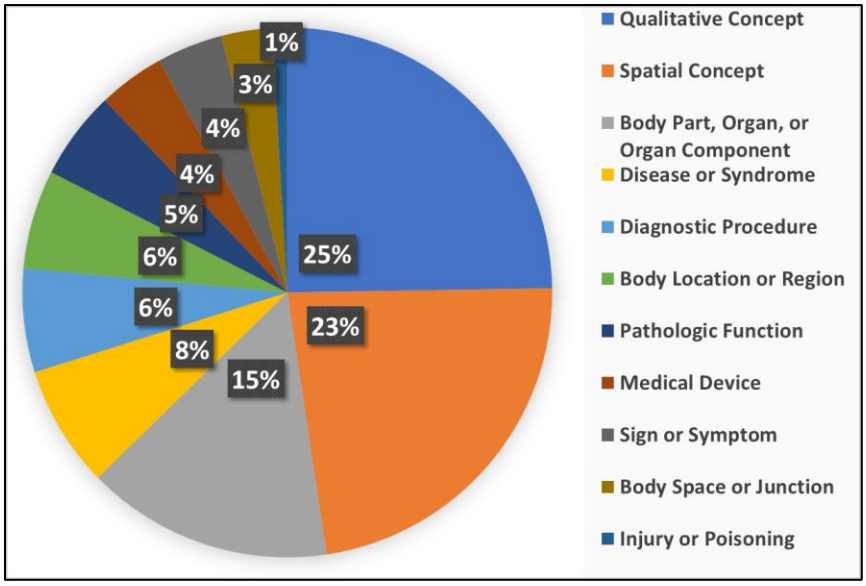

Figure 5. Distribution of all semantic types found.

While Figure 2 reveals most semantic types of interest, some minor semantic types cannot be ignored. For instance, neoplastic process and injury or poisoning can be extremely deadly if neglected. Also, medical devices installed such as Nasogastric tubes are indicators telling whether the patient's condition is critical.

Based on the statistic and the hierarchy of UMLS, original semantic types were group into six categories to simplify the outcome, which are given in Table 1. Tokens of semantic types found will all be classified according to this mapping.

Semantic Types	Abbr.	Group	Label
Body Part, Organ, or Organ Component	bpoc	1	A
Body Location or Region	blor		
Body Space or Junction	bsoj		

Injury or Poisoning	inpo	2	D
Pathologic Function	patf		
Disease or Syndrome	dsyn		
Sign or Symptom	sosy		
Neoplastic Process	neop		
Spatial Concept	spco	3	S
Qualitative Concept	qlco	4	Q
Medical Device	medd	5	M
Diagnostic Procedure	diap	6	P
Filter Word	oooo	0	O

Table 1. Mapping between the original semantic types and labels.

3.3.4 Aligning tokens with labels I

Instead of assigning semantic types to tokens of words, QuickUMLS returns the offsets of entities recognized. To generate labels for training, it is necessary to align label offsets with word tokens.

An approach utilizing the offsets of spaces in a sentence was developed as described in Figure 6.

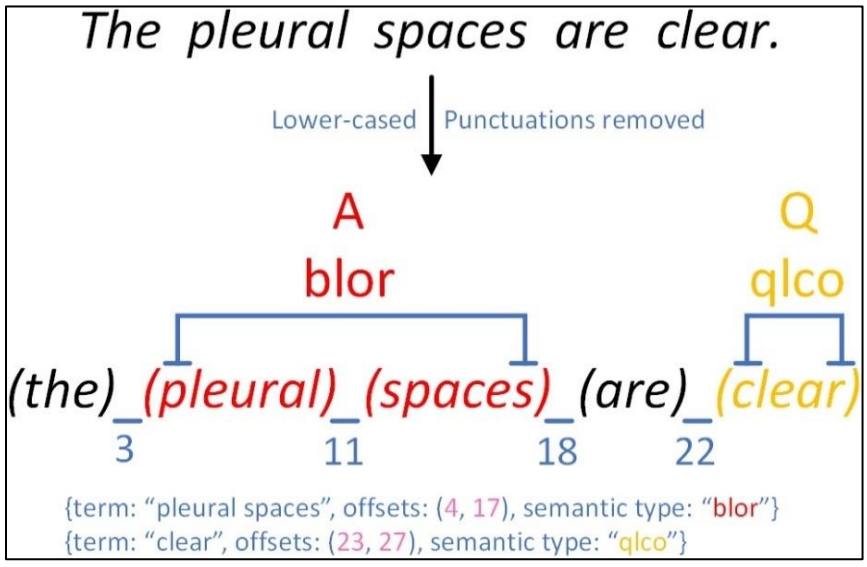

Figure 6. Aligning words with entities.

First, the input sentence is lower-cased and punctuation-removed. Then it is analyzed with QuickUMLS and entities retrieved. After that, the offsets of spaces are located, indicating the intervals of each word. At last, the offsets of each interval are checked against those of each entity. If the current word is contained by a entity, it is assigned with the label of that entity. Else, it is given the label

for filter words, O. This procedure is also applied to aligning words with classification results later.

3.4 BioBERT token classifier

3.4.1 Transformers and BioBERT

Recurrent neural networks, long short-term memory and gated recurrent neural networks have been established as efficient approaches in language modeling problems such as machine translation and named entity recognition. However, their performance is still far from satisfactory, as they cannot process inputs in parallel, and suffer from the loss of information, e.g., they cannot relate the current information with the previous one well.

To tackle these issues, attention mechanism was introduced and the most significant achievement is Transformers[19]. Transformers is an encoder-decoder architecture based on self-attention mechanism, and its encoder diagram is shown in Figure 7.

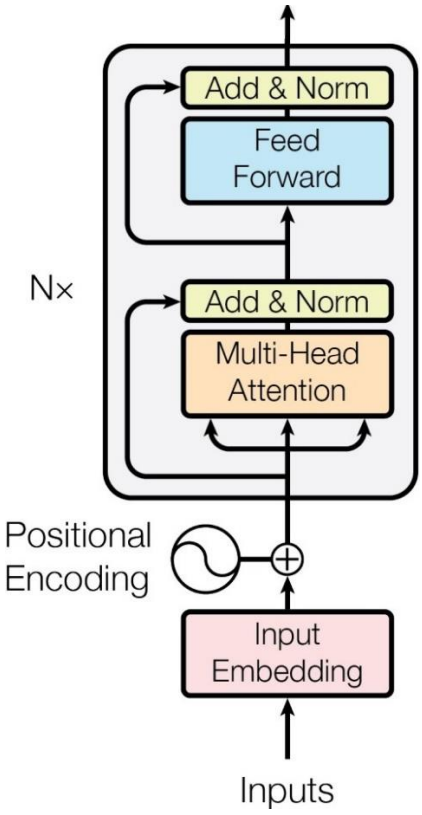

Figure 7. Transformers' encoder

While the standard Transformers architecture contains both the encoder and decoder, BERT only utilizes the encoder, which explains its full name, Bidirectional Encoder Representation from Transformers.

The key of this encoder is the self-attention mechanism, which adjusts the weight coefficient matrix according to the degree of association between words in the same sentence to obtain the representation of words:

$$Attention(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (1)$$

Among them, Q, K, V are word vector matrices, and d_k is the embedding dimension. The multi-head attention mechanism uses multiple different linear transformations to project Q, K, and V, and eventually concatenate attention results, as shown in formula (2) and (3):

$$MultiHead(Q, K, V) = Concat(head_1, \cdots, head_n)W^O \quad (2)$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \quad (3)$$

However, such an encoder structure does not have the ability to obtain the sequence of the entire sentence like RNN. To solve this problem, Transformers adds position encoding before data preprocessing, and sums it with the input vector data to obtain the relative position of each word in the sentence.

The fully connected feedforward network (FFN) in the Transformers structure has two dense layers: the first layer with an activation function of ReLU, and the second layer with a linear activation function. If the output of the multi-head attention mechanism is Z, and b is the bias vector, then FFN can be expressed as:

$$FFN(Z) = max(0, ZW_1 + b_1)W_2 + b_2 \quad (4)$$

3.4.2 Pre-training and fine-tuning

To train a BERT model for a specific dataset and task, two essential steps are required. One is pre-training, and the other is fine-tuning.

Pre-training also contains two objectives. The first is Multi-Mask Language Modeling, generally speaking, to randomly masked out 15% of words in the corpus and train the encoder stack and embedding layers to fill in the blank within a sentence. The second is next sentence prediction, which is to predict the next sentence of the input (Figure 8).

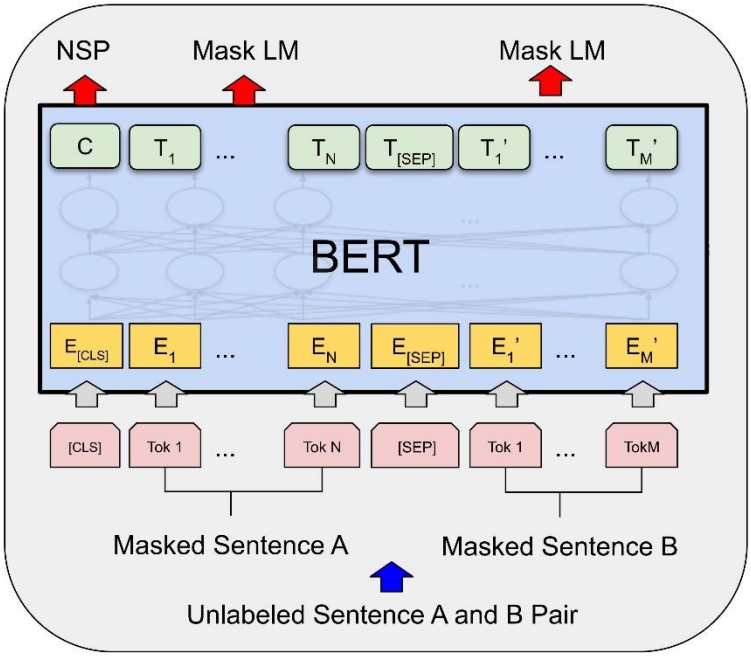

Figure 8. BERT pre-training.

As for our task, we used BioBERT[20], which was not only trained with texts from general domains, but also with corpora on the biomedical field such as PubMed and PMC.

Fine-tuning is configuring BERT for a specific type of task. While the structure of BERT is often explained with generative tasks like machine translation, it can be transferred to named entity recognition with a very simple change. As shown in Figure 9, changing the output from sentences to labels will transform the BERT from a translator to a classifier.

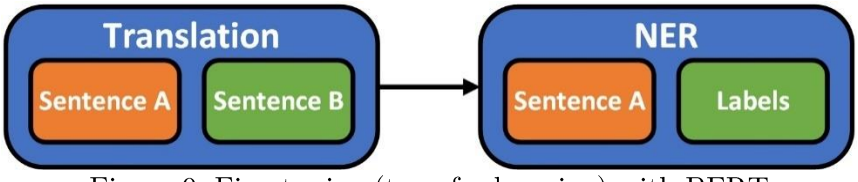

Figure 9. Fine-tuning (transfer learning) with BERT.

As for our task, we fed our data with labels generated using QuickUMLS as the training input to the default NER trainer

developed by Hugging Face Transformers[21] loading `biobert-large-cased-v1.1` pre-trained by Data Mining and Information Systems Lab at Korea University, with 10% as the test set and 20% in the training set for evaluation (Figure 10). After training, we built the model with Transformers' `AutoTokenizer` class and `AutoModelForTokenClassification` class with the BioBERT we just trained.

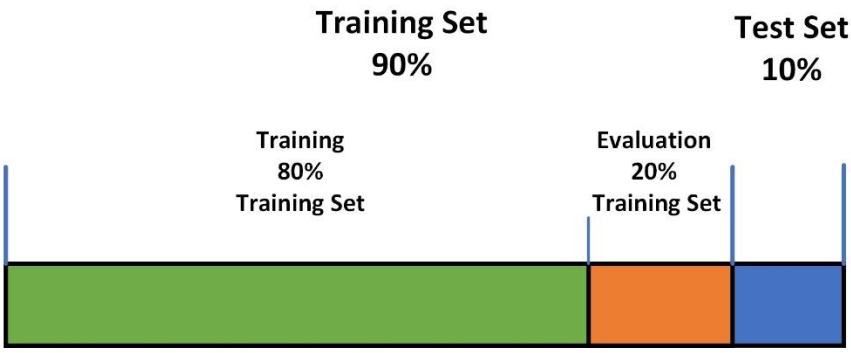

Figure 10. The distribution of test and training data.

3.4.3 Aligning tokens with labels II

Just like QuickUMLS, the output of the BioBERT token classifier is in the form of labels and offsets. However, the only difference lies in the integrity. A word in a sentence might be split into many tokens by the BERT tokenizer with their positions in the sentence recorded as offsets, as shown in Figure 11. Then the token classifier gives each token a label. But the `AutoTokenizer` class of Hugging Face Transformers does not provide built-in methods to concatenate tokens and labels together, making it hard to read and assess, so we came up with a simple algorithm to do so.

The breakthrough is again offsets. A word torn apart by the tokenizer has tokens with overlapping offsets. We only merged tokens with overlapping offsets, and merged their labels at the same time, as demonstrated in Figure 11. After the merging, we removed the separators at the beginning and the end of the sequence (`[CLS]` and `[SEP]`). Eventually, we obtained predictions easy to understand and evaluate.

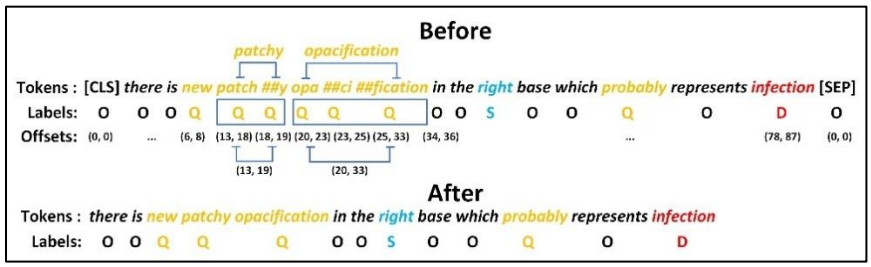

Figure 11. Merging and aligning tokens with labels.

3.5 Negation detection

Among all the entities detected, the D class requires extensive attention. In clinical history, it usually represents the observed symptoms, and most importantly, the goal of this radio examination, to find out whether the examinee carries a disease. In the main section, it is found associated with findings, and most importantly, the conclusion--whether the disease stated before is found.

To take advantage of the rigorous style of clinical reports, we implemented a simple rule-based method to detect negations related to the D class, as shown in Figure 12. First, we search the labels of a sentence to see if there exists Ds. If so, we look at tokens with O labels to see if there are negations using a dictionary. If we find a negation in the same sentence, it means the examinee does not carry the disease indicated by the label D. Else the examinee does, and a warning will be given out alongside the tokens of label D. Sample outputs of a sentence with negation and another without negation detection are given in Figure 13.

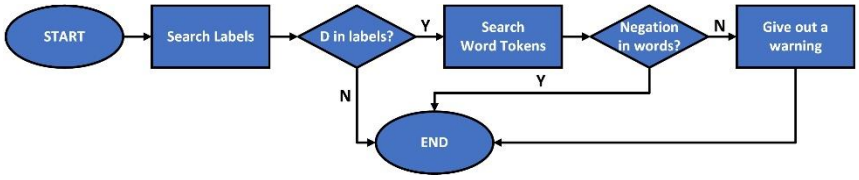

Figure 12. Negation detection.

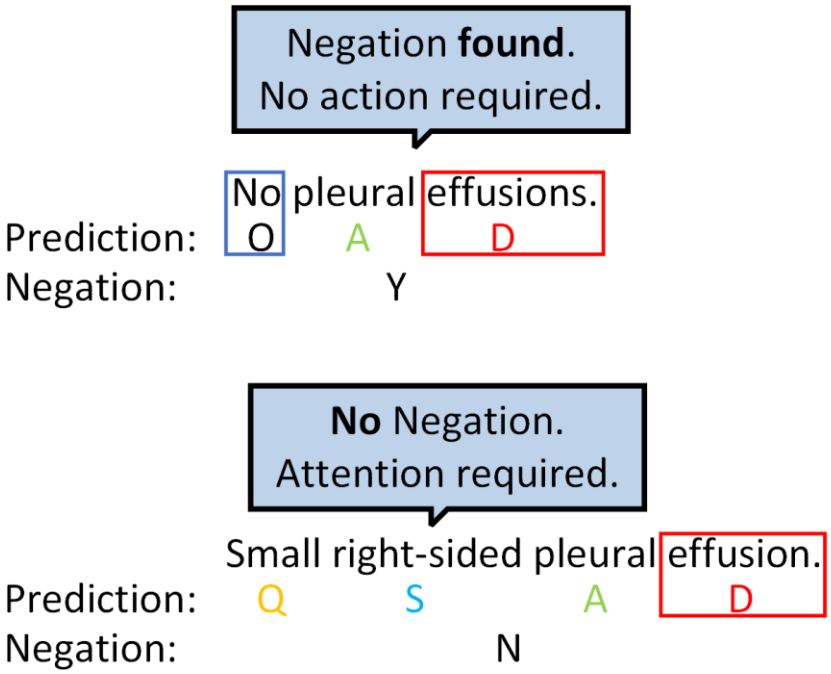

Figure 13. Sample outputs of negation detection.

4 Results and Discussion

4.1.1 Error analysis on NER

Given the method in which we obtained the annotation for training and the diversity of our labels, it is reasonable to evaluate whether the BioBERT predictor can imitate the QuickUMLS annotator well. Using the 10% test data (1058 reports) split before training, we calculated the F1 score and the strict accuracy for this goal, as shown in Table 2.

The definition of F1 score is given in (5), along with the definitions of precision and recall. As shown in formula (5), F1 score is a combination of precision and recall, assuming that they are equally important.

$$precision_k = \frac{TP}{TP+FP} \tag{4}$$

$$recall_k = \frac{TP}{TP+FN} \tag{5}$$

$$f1_k = \frac{2 * precision_k * recall_k}{precision_k + recall_k} \tag{6}$$

$$F1\ score = \frac{1}{n}\sum f1_k \tag{7}$$

F1 Score				
Label	Precision	Recall	F1-score	Support
A	0.7816	0.9793	0.8693	3431
D	0.5429	0.8870	0.6735	2070
S	0.7215	0.9663	0.8261	3172
Q	0.5505	0.9360	0.6933	3080
M	0.6583	0.8155	0.7285	645
P	0.6407	0.9554	0.7670	672
Micro Average	0.6491	0.9420	0.7686	13070
Macro Average	0.6493	0.9232	0.7596	13070
Weighted Average	0.6614	0.9420	0.7741	13070
Strict Accuracy				
Unbalanced		0.8483		
Balanced		0.9064		

Table 2. Results obtained with the test set.

As shown in the table, the recall of the BERT model is very high with an average over 90%, indicating that it can detect named entities well. On the contrast, the precision of all classes is not so satisfying as the recall, with an average of around 60%. To find out

the cause behind this phenomenon, we kept track of misclassifications and found several major types of errors.

While there exist several error types, not all of them are considered flaws. In fact, some of them can be seen as transcending the QuickUMLS annotator.

1) Connecting isolated concepts

Many errors are actually not errors, as the annotation generated by QuickUMLS are not so accurate, with an inability to comprehend the current and later concepts in some texts as a whole. In fact, many errors are corrections to the inability stated. For example, in Figure 14, the phrase "left side of the abdomen" was divided into two entities, a spatial concept and an anatomy concept, which is somehow correct but ignores the integrity of words. However, in the prediction, the same words were classified into an anatomy as a whole, which breaks the limitation of the QuickUMLS annotator.

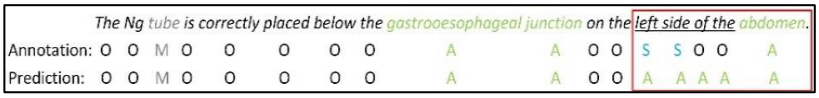

Figure 14. Breaking the limitation of the annotator.

2) Recognizing general terms

In radiograph reports, there are many general terms used as references of specific terms stated previously. For instance, at the beginning of a report, the writer might write down details of the film taken, like its types (X-ray, CT, etc.), body parts (anterior posterior, lateral, etc.), and that's also the only time in a scanning the token classifier should give "P" labels (diagnostic procedure) to words. Then, in the main section, the clinician will refer to the diagnostic procedure recognized with a highly generalized word "radiograph" or "film", which should not catch the reader's attention. However, in the real output, the token classifier often marks this generalized concept as a diagnostic procedure (Figure 15).

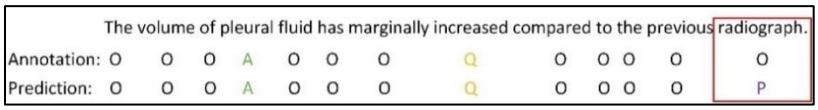

Figure 15. Recognizing general terms.

After observing some correct cases with diagnostic procedures at the end of a sentence, we found that it's these correct cases that lead the token classifier to this confusion. As displayed in

Figure 16, there are many lines describing the information of the examination, and most of them has the generalized term at the end, which is telling a specific type of examination as whole, but will mislead the token classifier to take all general terms at the end of a sentence as a diagnostic procedure worth noting.

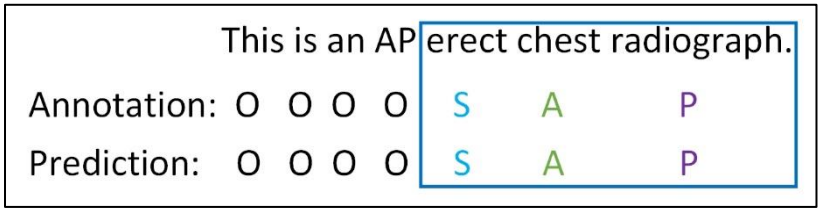

Figure 16. Correct cases that might cause misleading.

3) Fixing the errors caused by the annotator

Not only there exist invisible errors concerning contextual information as stated above, but also there are many noticeable misclassifications on isolated concepts missing important words and giving them the label "O". And the BioBERT token classifier can fix them well, like the case shown in Figure 17.

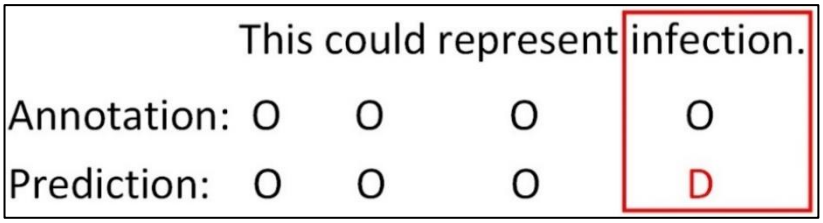

Figure 17. Fixing the errors caused by the annotator.

This also give us a point supporting that we use the BioBERT token classifier but not the QuickUMLS alone for the final output. The search algorithm used by QuickUMLS is not complete, which will make some mistakes at some time. And our BERT token classifier can serve as the last resort against these unnoticeable errors.

4) Inability to recognize abbreviations not in UMLS

One of the features of radiograph reports is the wide use of abbreviations. Not only there are many commonly recognized abbreviations shared by personnel across many different institutions and hospitals, but also even different doctors in a hospital might develop their individual styles of abbreviations, which bring huge difficulties for clinical NLP.

Such difficulties are so great that even such an intelligent model as BERT cannot deal with efficiently. While some common abbreviations were be recognized well like "CXR", others were never successfully classified correctly. For instance, as demonstrated in Figure 18, "AP" standing for "anterior posterior" is very common in the dataset and should be under the class "S", but you cannot find it with QuickUMLS annotator, and as result, the BioBERT token classifier could not recognize a single "AP" in the entire training and evaluation process. This could be resolved by manually annotating these abbreviations, given more time and resources.

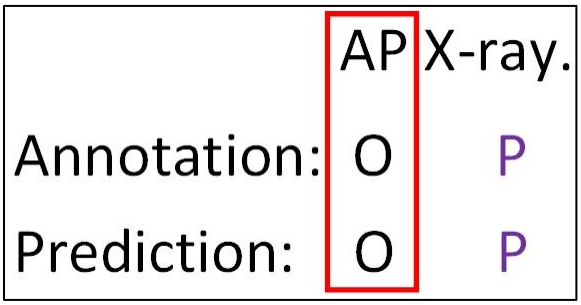

Figure 18. Incorrectly classified abbreviations.

4.1.2 Error analysis on negation detection

The dataset for evaluating the negation detection was formed by manually selecting 100 sentences with label "D" and negation and 100 sentences with label "D" but no negation. Then these samples were scrambled and detected for negation according to the procedure previously describe in Figure 1. Be advised that only sentences without any negation detected will be tagged positive.

And the result is given in table 3. Thanks to the rigorous style of radiograph reports, a simple algorithm as the one proposed by us could achieve good results.

		Prediction	
		Positive	Negative
Truth	Positive	79	21
	Negative	17	83

Table 3. The confusion matrix of negation detection.

After looking into the error cases, we found two main reasons behind them.

1) No "D" in the sentence

While QuickUMLS has a database to refer to and BioBERT has embedding layers, a human annotator relies on his/her own knowledge to determine whether there are disorders in a sentence most of the time, which may lead to errors. For instance, "consolidation" was recognized by us as a disorder, but for the token classifier, it goes under the qualitative concept, which would not even trigger the negation detector.

2) Inability to detect implicit negations

Another major cause for errors is our rule is too primitive. It was designed to detect explicit negations, like no, not, isn't, etc. When encountering implicit negations embedded in other qualitative concepts, like unremarkable, it would ignore it, take it as a positive, and give out an alarm.

In a word, while the high standard and rigorous style of reports help with most of the cases, there are still a considerable number of cases where a more sophisticated rule should be explored and implemented.

5 Conclusion

To tackle the task of named entity recognition for specific entities on radiograph reports, we first conducted feature extraction and annotation with UMLS. Then with the annotation acquired, we fine-tuned a BioBERT token classifier, evaluated it against the QuickUMLS annotator, and obtained a satisfying result. The biggest advantage of our is that BioBERT can learn the contextual syntactic structure and the semantic information of the context, making it perform better than other models. Apart from this, not only it is much smaller than the QuickUMLS annotator in volume, but also it can get rid of the innate inability of failing to comprehend the contextual information of the former. Beyond these, a simple yet efficient negation detection mechanism for disorders was developed, which can give out warning information when a disorder is confirmed in a sentence. Further improvements could be made on refining the rules of negation detection and deploying it as a web service.

Reference

[1] Küçük D, Yazici A. Rule-based named entity recognition from Turkish texts[A]. Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications[C]. 2009: 456-460.

[2] Kim J H, Woodland P C. A rule-based named entity recognition system for speech input[A]. Sixth International Conference on Spoken Language Processing[C].2000.

[3] Chandel A, Nagesh P C, Sarawagi S. Efficient batch top-k search for dictionary- based entity recognition[A]. 22nd International Conference on Data Engineering (ICDE'06)[C]. IEEE, 2006: 28-28.

[4] FRIEDMAN C, HRIPCSAK G, DUMOUCHEL W, et al. Natural language processing in an operational clinical information system [J]. Natural Language Engineering, 1995, 1(1):83-108.

[5] SEVENSTER M, VAN O R, QIAN Y. Automatically correlating clinical findings and body locations in radiology reports using MedLEE [J]. Journal of Digital Imaging, 2012, 25(2):240-249.

[6] Bodenreider O. The Unified Medical Language System (UMLS): integrating biomedical terminology. Nucleic Acids Res. 2004 Jan 1;32(Database issue):D267-70. doi: 10.1093/nar/gkh061. PubMed PMID: 14681409; PubMed Central PMCID: PMC308795.

[7] CIMINO J J, CLAYTON P D, HRIPCSAK G, et al. Knowledge-based approaches to the maintenance of a large controlled medical terminology [J]. Journal of the American Medical Informatics Association, 1994, 1(1):35-50.

[8] LI Y, GORMAN S L. Section classification in clinical notes using supervised hidden markov model [C]// ACM, 2010:744-750.

[9] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks[J]. science, 2006, 313(5786): 504-507.

[10] COLLOBERT R, WESTON J, KARLEN M, et al. Natural language processing (almost) from scratch [J]. Journal of Machine Learning Research, 2011, 12(1):2493-2537.

[11]SOCHER R, HUVAL B, MANNING C D, et al. Semantic compositionality through recursive matrix-vector spaces [C]// Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, 2012:1201-1211.

[12]LAMPLE G, BALLESTEROS M, SUBRAMANIAN S, et al. Neural Architectures for Named Entity Recognition [J]. 2016:260-270.

[13]Devlin J, Chang MW, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv: 1810.04805, 2018.

[14]Strakova J, Straka M, Haji6 J. Neural architectures for nested NER through linearization. arXiv preprint arXiv: 1908.06926, 2019.

[15]Luca Soldaini and Nazli Goharian. "QuickUMLS: a fast, unsupervised approach for medical concept extraction." MedIR Workshop, SIGIR 2016.

[16]Okazaki, Naoaki, and Jun'ichi Tsujii. "Simple and efficient algorithm for approximate dictionary matching." COLING 2010.

[17]S. Sarawagi and A. Kirpal. Efficient set joins on similarity predicates. In SIGMOD, 2004.

[18]D. A. Grossman and O. Frieder. Information Retrieval: Algorithms and Heuristics. Springer, 2012.

[19]Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., … Rush, A. M. (2020). Huggingface's transformers: State-of-the-art natural language processing. arXiv:1910.03771 [cs]. http://arxiv.org/abs/1910.03771

[20]Vaswani, A., Shazeer, N.M., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., & Polosukhin, I. (2017). Attention is All you Need. ArXiv, abs/1706.03762.

[21]Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, Jaewoo Kang, BioBERT: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics, Volume 36, Issue 4, 15

February 2020, Pages 1234–1240, https://doi.org/10.1093/bioinformatics/btz682