Master Thesis

Chinese Healthcare Named Entity Recognition (CHNER) Using BiLSTM-CRF Classifiers

L. Ma

a thesis submitted in partial fulfilment of the requirements for the degree of

MA Linguistics
(Text Mining)

Vrije Universiteit Amsterdam

Computational Lexicology and Terminology Lab
Department of Language and Communication
Faculty of Humanities

Supervised by:	Luis Morgado da Costa
2^{nd} reader:	Vliet, H.D. van der
Submitted:	May 28, 2024

Abstract

This thesis describes an experiment carried out based on the ROCLING 2022 Shared Task, where the problem of Chinese Healthcare Named Entity Recognition (CHNER) was proposed. This thesis covers different aspects of the experiment, including background information, methodology, experiment results, conclusion and discussion.

During the experiment, we incorporated *Jieba* supported by different dictionaries to deploy different segmentation strategies to the same Chinese Healthcare Named Entity Recognition (CHNER) datasets. We used the default dictionary of *Jieba* and dictionaries that borrow lexicon from external domain corpora (e.g., the CBLUE datasets) to create three word-based versions of the CHNER datasets (train-dev-test data). The train-dev sets (include gold labels) were used to train the BiLSTM-CRF classifiers. We also implemented matching word embeddings (WEs) by applying the corresponding segmentation scheme on the training corpus (train-dev data of the CHNER datasets) for vector representation.

We trained a Baseline model (character-based) that uses characters of the original datasets. We also trained three word-based systems (Jieba Base, Jieba Upgrade, Jieba Full) that use sub-tokens bigger than characters. Finally, we picked the best word-based and character-based models to predict the labels of the test set. The test data (exclude gold labels) was processed by applying the same segmentation strategy of the training (train-dev) data.

Through comparing the performance of the two models (word-based model versus character-based model), the thesis aims to explore the influence of different Word Segmentation strategies on the Chinese Healthcare Named Entity Recognition (CHNER) task. Our best system (Baseline) achieved a 0.676 and 0.612 f1 score on the development set and the test set.

Declaration of Authorship

I, Long Ma, declare that this thesis, titled *Chinese Healthcare Named Entity Recognition (CHNER) Using BiLSTM-CRF Classifiers* and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a degree degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: 28.May.2024

Signed:

Acknowledgments

Throughout my thesis project, I have received immense support and help from school teaching faculties and technicians. I would like to offer my thanks to Dr. Luis Morgado da Costa from Vrije Universiteit Computational Linguistics and Text Mining Lab (CLTL), who has been continuously supporting me throughout the project. He has provided humongous help with the problems during the composing of thesis and offered meticulous review of my work, which has greatly elevated my writing of the master's thesis.

Moreover, I am extremely grateful for his guidance and assistance at every stage of the process, especially during the planing and final checking phase. Secondly, I would like to extend my heartfelt thanks to the second reader Dr. Vliet, H.D. van der from Vrije Universiteit Computational Linguistics and Text Mining Lab (CLTL).

Without their support and supervision, my project would not have been the same. I am truly thankful for their contributions.

Long Ma

List of Figures

Contents

Chapter 1

Introduction

In the healthcare domain, patients often seek information for self diagnosis before consulting a doctor. Web texts, as the the primary resource that most patients refer to, has become quite popular. With the help of search engines, patients are free to look for information via various media, such as health news, digital health magazines, medical question-answer forums.

However, the raw text from these sources often contains many terms that are primarily identified as healthcare entity names that are used in the specific domain. For example, symptoms (e.g., “血壓” (xuè yā) (“blood pressure”)), body parts (“基因組” (jī yīn zǔ) (“genetic group)), and so on. It is difficult for people with little or no field knowledge to understand or learn the complicated expressions. Therefore, there has been a growing public interest in developing technologies that extract healthcare knowledge (named entities) to help ease the information gap between patients and doctors.

1.1 Problem setting

Named entity recognition (NER) has been studied extensively due to its wide range of applications in various fields, including the healthcare domain. NER is a Natural Language Processing (NLP) task that aims to locate and identify mentions of named entities (e.g., person, organization, and location) in written texts. Named entities (NEs) are words or phrases that carry key information in a certain topic (Mohit, 2014). For example, in the sentence “[ORG U.S.] president [PER Joe Biden] gave a speech in [LOC Alaska].” “U.S.” is an organization, “Joe Biden” is a person, and “Alaska” is the mention of a location.

Chinese healthcare named entity recognition (CHNER) is a use case of the NER technology that extracts meaningful knowledge in the healthcare domain from Chinese text (Cheng et al., 2021).

Retrieving healthcare entity information is of great value to various medical applications such as clinical decision support systems (Wu et al., 2015) and automated medical coding (Gong et al., 2020). Another example is a CHNER system provides information that essentially aids patients with finding self-aid solutions and receive treatments remotely.

Conventional CHNER technologies treat CHNER as a sequence labeling problem since both the boundaries and the labels (categories) of entities are predicted at one go. Traditional Machine learning methods use language models such as Hidden Markov

Models (HMM), Conditional Random Field (CRF) for extracting healthcare entities. More recently, neural networks such as BiLSTM and BERT (transformers) have demonstrated even better performance in predicting the healthcare labels.

Inspired by Lee and Lu (2021)'s research on CHNER, the ROCLING 2022 Shared Task (Lee et al., 2022b) aims to extract not only the category of the text (e.g., "SYMP"), but also the BIO notation (i.e., "B", "I", "O") that aligns with the text. For example, in "維持体液电解质的平衡" (Wéi chí tǐ yè diàn jiě zhì de píng héng) ("Maintaining the balance of body fluids electrolytes"), "体液" (tǐ yè) ("body fluid") and "电解质" (diàn jiě zhì) ("electrolytes") are "BODY" entities with different BIO notations. "体液" (tǐ yè) ("body fluid") is labeled as "B-BODY" ("B" indicates being at the start position of a "BODY" entity) and "电解质" (diàn jiě zhì) ("electrolytes") is labeled as "I-BODY" ("I" indicates being inside of a "BODY" entity).

Mining Chinese healthcare named entities concerns the word segmentation problem since the segmentation scheme determines the entity boundaries. For example,

```
Text:
"电解质" (diàn jiě zhì) (`"electrolytes")

Characters:
"电" (diàn) ("electricity")
"解" (jiě) ("release")
"质" (zhì) ("essence")

Words:
"电" (diàn) ("electricity")
"解质" (jiě zhì) ("released items")
```

The sample shows different ways of segmenting "电解质" (diàn jiě zhì) ("electrolytes"). When we segment the text into "电" (diàn) ("electricity"), "解质" (jiě zhì) ("released items"), the fragments should be treated as two singular words by the CHNER classifier. Hence the chances of the two parts being classified as different entities increases. Based on such intuition, the word segmentation unavoidably influences the performance of CHNER classifier(s).

1.2 Research Question

This paper sets out to determine:

```
How do character-based segmentation techniques compare to word-based
segmentation techniques (enriched by domain-specific word boundaries)
influence the task of CHNER?
```

In other words, we explore which segmentation strategy (word-based versus character-based) helps the same language model (BiLSTM-CRF) obtain better CHNER results on the same data.

1.3 Method

Motivated by prior work, we deployed the data (train, development, test data) and evaluation metrics of the shared task.

Each system utilizes data and meta data (annotations), with matching word embeddings (WEs) to train the same base model (BiLSTM-CRF).

We established a character-based system (Baseline) and three word-based systems (Jieba Base, Jieba Upgrade, Jieba Full.)

The Baseline system uses characters and gold labels, e.g.,

```
"體" (tǐ) ("body") "B-BODY"
"細" (xì) ("slim") "I-BODY"
"胞" (bāo) ("flesh") "I-BODY"
```

The character-based classifiers were trained on characters with corresponding gold labels (train-dev set).

Meanwhile, the word-based systems (Jieba Base, Jieba Upgrade, Jieba Full) use *Jieba* supported by different dictionaries (Default, ROLING Char, FUSION) for three additional versions of the CHNER datasets (train-dev-test). The word-based system uses words (or sub-tokens) and merged gold labels, e.g.,

```
"體細胞" (tǐ xì bāo) ("cell of body") "B-BODY"
```

By principle, *Jieba* segment the text into pieces larger than characters based on the restricted lexicon. In the example, “體細胞” (tǐ xì bāo) (“cells of body”) was treated as one “B-BODY” entity since the vocabulary is defined in the dictionary. The gold labels of the segmented text (train-dev data) are merged to align with the updated word boundaries. The test data was processed by applying the matching segmentation strategies of the training data (train-dev set.) The gold labels of the test data remain untouched for final evaluation.

Based on the segmentation schemes, we created three additional versions of the original datasets. The input text was transformed into word vectors by matching WEs (segmentation scheme) for training the BiLSTM-CRF classifiers.

We compared the validation of the classifiers on the development data over 3, 5 and 7 epochs to find the best models. We selected the best word-based system to compare with the character-based system and conclude our experiment based on their performance (f1 scores) on the test data.

1.4 Outline

This section outlines how the remainder of the thesis is organized. In the next chapter, Chapter 2, I will introduce the background information concerning the research topic. I first elaborate on the task definition and related work, followed by introduction on the Chinese Word Segmentation (CWS) problem in relation to the CHNER study. The external resources used in our experiment are also discussed.

In Chapter 3 I will explain how the experiment is carried out in two steps: Data pre-processing; Training and hyper-parameter tuning the classifiers.

Chapter 4 focuses on presenting the experiment results. The first part explains the evaluation metrics adopted by the experiment. Part two (development data) and part three (test data) look closely into the specifics of the validation results.

Chapter 5 provides details concerning the classification errors, with breakdowns of the two selected systems, i.e., the Baseline model (character-based, Epoch=3) and the Jieba Full model (word-based, Epoch=5) across different healthcare entity categories.

Chapter 6 returns to the research question and provides the answer with discussions (limitations and reflection on future work).

Chapter 7 summarizes the most important information of our experiment.

Chapter 2

Background Information

2.1 Chinese Healthcare Named Entity Recognition

Chinese Healthcare Named Entity Recognition (CHNER) is an use case of **Named entity recognition (NER)** technology that is apt for the Chinese healthcare domain. **Named Entity Recognition (NER)** is a natural language processing (NLP) task that falls in the broader scope of Information Extraction (IE). The task aims to locate and identify mentions that fit a collection of predetermined definitions that distinguish named entities (NEs) from normal text in the task-specific domain. (Li et al., 2020).

Named entities (NEs) are words or phrases with specific meanings on a certain topic (Mohit, 2014). NEs can be divided into generic NEs and domain-specific ones. Generic NEs are mentions that are meaningful across languages and research purposes. Some most common generic NEs are person names, places, organizations, etc. For example, “Xi” (person name), and “Wenzhou” (location) are different entities in the sentence “President [PER Xi] gave a speech in [LOC Wenzhou].” Meanwhile, domain-specific NEs refer to entities that are closely linked to a specific domain study, e.g., the English word “Katrina” is commonly recognized as a a natural disaster name in the geometric field. However, it is commonly treated as a person name more globally.

NER technologies are applied widely among NLP downstream applications, such as question answering (QA), machine translation (MT) and entity relation extraction. The information retrieved can be utilized for various computational purposes (Li et al., 2020). For example, a machine translation system separates entity from normal text in sentence “[President Bush PER] gave a speech”. “President Bush” is a “PER” mention that is translated as the person name, which is an use case of NER knowledge that helps with the MT system.

This thesis focuses on extracting the **Chinese Healthcare named entities (CHNEs)**, which are words or characters with special meanings in the Chinese healthcare domain. The most common healthcare names are diseases, symptoms, medical procedures, medications, anatomical terms, e.g.,

```
"主要症狀包括頭痛、嘔吐，確診為腦出血。”
(zhǔ yào zhèng zhuàng bāo kuò tóu tòng, ǒu tù,
què zhěn wèi nǎo chū xiě .)
("The main symptoms includes headache, vomiting,
and was diagnosed as cerebral hemorrhage.")
```

There are three mentions of entities named as symptoms in the text: “頭痛” (tóu tòng) (“headache”), “嘔吐” (ǒu tù) (“vomiting”) and “腦出血” (nǎo chū xiě) (“cerebral hemorrhage”), they can all be labeled as “SYMP”.

CHNER technologies provide insightful information for various applications such as entity relation extraction, medical question-answering, and building clinical knowledge graph. For example, descriptions such as “頭痛” (tóu tòng) (“headache”), “嘔吐” (ǒu tù) (“vomiting”) are very insightful for establishing the mapping between clinical records and clinical diagnostic codes. Extracting CHNEs facilitate interoperability and transparency in healthcare data management and clinial analysis (Cheng et al., 2021).

2.2 Prior Work

While there has been many rule-based approaches that attempt to utilize heuristic rules and Lexicons for extracting the healthcare entities, the retrieval of named entities faces two main challenges: (1) recognizing named entity boundaries; (2) identifying named entity categories (classes). The two pointers are normally addressed simultaneously since NER problems are often addressed as sequence labeling tasks.

Based on Li et al. (2020)’s survey, existing NER methodologies could be grouped into four streams: (1) Rule-based techniques that relies on manually crafted rules; (2) Unsupervised learning functions that employ algorithms to utilize unlabeled training data; (3) Supervised learning techniques that are empowered by features engineering; (4) Deep learning (DL) methods that deliver classification and (or) detection directly from raw data. We provide more details about the supervised learning and deep learning approaches since they are the most commonly applied technologies in NER researches.

The effectiveness of Supervised machine learning (SML) techniques on NER problems are well documented. Traditional SML requires large amount of annotated data to prepare the language model for predicting on new data. Moreover, SML models rely on feature engineering to create meaningful abstraction of the text that are “friendly” for ML algorithms to capture input features. According to Nadeau and Sekine (2007), the SML features are often in the forms of Boolean, numeric, nominal values. These values are used to encode information such as the capitalization pattern, the length, and the lower-case of the current text.

Crafting NER features requires mining new features or modifying existing features based on linguistic intuition and statistic analysis. Nadeau and Sekine (2007) categorizes ML features into three groups based on observations of NER studies from 1991 to 2006. They are: Word-level features, list lookup features, and document (or corpus) features.

Word-level features entail different aspects concerning the composition of individual words, such as casing, punctuation mark, and presence of numeric content. List lookup features involve look-up systems such as “gazetteer”, “lexicon”, “dictionary” that are task-specific. Machines use these domain lexicon for reference. Document features encompass meta-information about the document and statistical characteristics of the corpus. Document features are typically derived from both the content and structure of the document.

However, the development of supervised ML was burdened by the bottleneck of feature engineering and pipeline cascading errors. **Recurrent neural networks (RNNs)** alleviated the problem by mimicking the cerebral cortex (neurons) of human brains. RNNs are an extension of traditional Feed-Forward neural networks designed

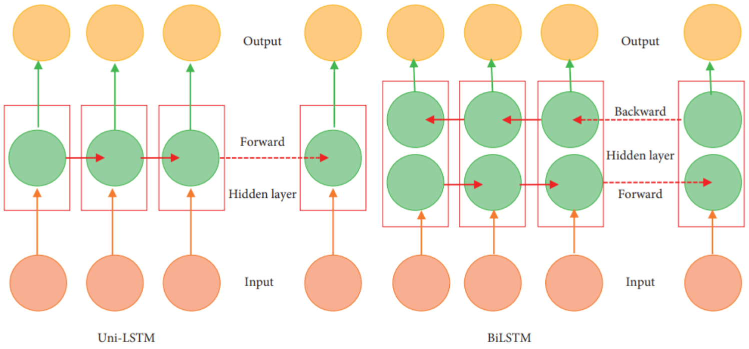


Figure 2.1: Uni-LSTM/BiLSTM architecture (picture based on Siami-Namini et al. (2019))

to handle variable-length sequence inputs. The technology retains information from previous inputs to handle sequences with varying lengths to forecast on sequential series data. Such networks involve the usage of sigma cells characterized by a recurrent hidden state layer that is influenced by previous states at each time step.

However, conventional RNNs suffers from information loss led by information exchange across multiple layers and excessively large gradients that complicates the training process (Siami-Namini et al., 2019). *Long Short-Term Memory (LSTM)* networks addresses the shortcoming by expanding the memory capacity. LSTM based models use ”gated” cells to preserve or delete information selectively based on the weight values assigned during training (Yu et al., 2019).

Uni-directional Long Short-Term Memory (uni-LSTM) networks is a example of such design, which typically involves four types of gate cells: (1) Forget gate, which deploy a sigmoid function to determine the amount of previous memory that needs to be forgotten. Such algorithm uses a value between 0 and 1 to indicate the level of information relevance, which allows the network to discard outdated information; (2) Input gate, which controls the flow of new information into the memory cell through regulating the size of new content added to the memory. A “sigmoid” layer and a “tanh” layer are used for decision of the input updates and generating new candidate values of memory incorporation; (3) Output gate, which modulates the amount of information that is output from the memory cell. The gate first employs a “sigmoid” layer to select relevant memory components before applying a “tanh” function for non-linear mapping. Followed by multiplication with the output of another “sigmoid” layer; (4) Cell Activation Vector, which comprises of two components −the partially forgotten previous memory and the modulated new memory. This vector represents the current state of the memory cell. Such architecture provides more streamlined information flow and more efficient memory storage in processing sequential data (Liu and Guo, 2019).

Bidirectional LSTM (BiLSTM) networks, or deep bidirectional Long Short-Term Memory (BiLSTM) networks is another variation of the RNNs. BiLSTM-based models establish flat NER layers stacking that utilizes multiple independent bidirectional LSTM units simultaneously to extract nested entities. Such architecture leverages information from both past and future time steps in RNNs to deal with more complex tasks, such as NER. Through updating the bidirectional weight dynamically, BiLSTM based models capture bidirectional temporal dependencies that encode more advanced linguistic context (e.g., polysemy) and patterns of the input data (e.g., semantics and syntax) (Siami-Namini et al., 2019).

Figure 2.1 demonstrates how LSTM-based and BiLSTM-based networks function. An uni-LSTM networks (on the left) processes each time step of the input sequence in a one-way manner (from past to future). This means the hidden state of the networks at each step is only dependent on the preceding time step.

In contrast, a BiLSTM networks handles input simultaneously in a bi-directional way (from past to future and from future to past both.) The dual directional information collection allows capturing the context from both sides of the input. The BiLSTM models pertain to the advantage of extracting long-range dependencies from the text. Therefore, Bi-LSTM networks exhibit greater resilience in vanishing gradient problems since information from both past and future states are both available during the training (Siami-Namini et al., 2019).

Moreover, standard Bi-LSTM based models are often used in combination with **word embeddings (WEs)**. WEs, or Distributed representations of words, are dense vectors in a high-dimensional space that serve as the abstract of the text. It involves forming a real-valued vector representation of the input from a predefined corpus with fixed amount of vocabularies (Noraset et al., 2017). The dense vector representation projects insight on word similarities since semantically similar words such as synonyms or words of the same category are closer in the vector space (distributional hypothesis). WEs have shown great potential in capturing both the syntactic (lexical) and the semantic aspects of text. The vector distance between the embeddings is useful for measuring the analogical relationships between words. A well-known example is the similarity between ”king” and ”queen” are high since the terms share a close semantic relation.

2.3 ROCLING 2022 Shared Task

ROCLING 2022 Shared Task aims to develop healthcare entity extraction technologies that are useful for the Chinese medical Named Entity Recognition (CHNER) task proposed by Lee and Lu (2021). The task deployed the same entities of the Chinese HealthNER Corpus (Lee and Lu, 2021) along with the “IOB2” notation put forward by Ramshaw and Marcus (1999). Each unit in the text is assigned a tag to represent the (non) entity type along with the NER boundary, i.e., “B” implies the start of a named entity, “I” is the non-start indicator of the character being inside an entity, “O” includes all other cases.

Table 2.1 shows the 10 types of healthcare entities listed in the shared task. The abbreviations were used to represent different healthcare genres, e.g., “CHEM” is short for chemicals, which encompasses the text belonging to a chemical entity used to treat the human body. Besides, 21 labels were assigned by the shared task for labeling of the medical text. They are “B-EXAM”, “B-BODY”, ”B-DISE”, ”I-DISE”, “B-SYMP”, “B-TREAT”, “B-CHEM”, “I-CHEM”, “I-SYMP”, “B-TIME”, “B-SUPP”, “I-BODY”, “I-TREAT”, “B-INST”, “B-DRUG”, “I-DRUG”, “I-TIME”, “I-INST”, “I-SUPP”, “I-EXAM”, “O”. Each label contains two pieces of information: entity boundary and entity type. For example, “B-BODY” represents the current text is positioned at the start a “BODY” entity. The “IOB2” notation and the entity type are separated by a “-” delimiter.

The *ROCLING 2022 Shared Task* (Lee et al., 2022a) uses the **Chinese Healthcare**

Entity	Definition	Example
BODY	Physical structure of human or animal	“腎” (shèn) (“kidney”)
SYMP	Any physical or mental change due to disease	“痒” (yǎng) (“itchy”)
INST	Any tool or device for medical performance	“刀” (dāo) (“knife”)
EXAM	The act of examining or checking for diseases	“檢查” (jiǎn chá) (“test”)
CHEM	Any chemical element from the human body	“素” (sù) (“element”)
DISE	An illness caused by health problems	“流感” (liú gǎn) (“flu”)
DRUG	Any natural or artificial chemical in medicine	“草” (cǎo) (“herb”)
SUPP	Supplement used for health improvement	“魚油” (yú yóu) (“fish oil”)
TREAT	A method of behavior used to treat diseases	“術” (shù) (“surgery”)
TIME	Time span of existence	“孕” (yùn) (“pregnancy”)

Table 2.1: Entity type Description

Named Entity Recognition (CHNER) datasets[1], which was created based on the Chinese HealthNER corpus. The NYCU NLP Lab (Lee and Lu, 2021) collected and annotated the Chinese HealthNER corpus.

The corpus mainly comprises of micro-blog text crawled from online healthcare websites, news, and medical forums. Raw data contains many online jargon or colloquial expressions from conversations between doctors and patients, e.g., “拉稀” (lā xī) (“diarrhea”), “腹瀉” (fù xiè) (“diarrhea”), “竄稀” (cuàn xī) (“diarrhea”) all refer to the same clinical symptom of diarrhea. The crawled data was further filtered after removing content such as HTML tags, images, videos and embedded web advertisements, the selected sentences were randomly sampled to obtain a more diverse data collection. The size of the Chinese HealthNER corpus was further expanded through search query on Chinese Wikipedia (zh_TW version) using the existing healthcare entities.

Finally, the processed data was split into train-dev-test files (referred to as “train file”, “dev file”, “test file”, “truth file”) for training and evaluation. In total, the datasets contain 30,692 sentences or around 1.5 million characters with 68,460 samples of 10 distinct named entity types.

Three undergraduate students with Chinese language background participated in annotations (both word segmentation and named entity tagging.) The inter-annotator agreement (IAA) of both tasks reached an overall 84.1%. More demographic details about the annotators is unknown. Table 2.2 provides an overview of the entity types (names of CHNER entities) and their distributions across the train and evaluation datasets. Predominantly, the majority of entities are “BODY”, exceeding one third of the entire database. Followed by “SYMP”, “DISE” and “CHEM” entities. These entities together account for 82% of the all healthcare entities. Meanwhile, “INST”, “SUPP”, “TIME”, “DRUG”, and “EXAM” entities appear much less frequently, taking an averaged 3.5% of the data. Table 2.1 provides more details concerning the definition of the entities with examples.

Regarding the data format, the train-dev files contain data and metadata saved in the format below:

```
{
```

[1] https://github.com/NCUEE-NLPLab/Chinese-HealthNER-Corpus

Entity Type (Tag)	Train(%)	Test(%)
Body (BODY)	26,411(38.58%)	5,315(39.76%)
Symptom (SYMP)	12,904(18.85%)	1,944(15.54%)
Instrument (INST)	1,089(1.59%)	250(1.87%)
Examination (EXAM)	2,622(3.83%)	207(1.55%)
Chemical (CHEM)	6,834(9.98%)	1,718(12.85%)
Disease (DISE)	10,079(14.72%)	2,609(19.52%)
Drug (DRUG)	2,225(3.25%)	481(3.60%)
Supplement (SUPP)	1,525(2.23%)	183(1.37%)
Treatment (TREAT)	3108(4.54%)	468(3.50%)
Time (TIME)	1,663(2.43%)	194(1.44%)
Total	68,460(100%)	13,369(100%)

Table 2.2: Named Entity Distribution

```
"id": 00002,

"genre": text genre, i.e., "ft" as formal texts, "sm" as social media,

"sentence": "治療胃病？" (zhìliáowèibìng?)  (treating stomach disease?),

"word": [ "治療", "胃病", "？"]
(zhìliáo wèibìng ?) ("treat", "stomach disease", "?"),

"word_label": ["O", "DISE", "O"],

"character": [ "治", "療", "胃", "病", "？"]
(zhì liáo wèi bìng ?) ("treat", "heal", "stomach", "disease", "?"),

"character_label": ["O", "O", "B-DISE", "I-DISE", "O"]
}
```

7 key-value pairs are used to represent different aspects of the text: “id” is the numeric identifier (index) of the sentence, e.g., “00002” indicates the index of the sentence is “00002”. “Sentence” contains the complete text of the sample. “word” preserves the sub-tokens of the sample (smaller chunks rather than the whole sentence), while “character” records the sentence input as characters. “Word_label” stores (non-BIO) tags of the corresponding chunks, which is different from “character_label”, since “character_label” adopts the “BIO2” notation. “Genre” indicates the source of the data, i.e., “ft” shows the nature of formal texts, while “sm” shows the text is from social media.

```
Test file:
"治" (zhì)  ("treat")
"療" (liáo)  ("heal")
"胃" (wèi)  ("stomach")
"病" (bìng)  ("disease")
"？" ("?")
```

```
"拉" (lā)  ("draw")
"稀" (xī)  ("liquid")
"腹" (fù)  ("stomach")
"瀉" (xiè)  ("pour")
"!" ("!")

Truth file:
"治" (zhì)  ("treat") "B-TREAT"
"療" (liáo)  ("heal") "I-TREAT"
"胃" (wèi)  ("stomach") "B-DISE"
"病" (bìng)  ("disease") "I-DISE"
"？ " ("?")

"拉" (lā)  ("draw")  "B-DISE"
"稀" (xī)  ("liquid") "I-DISE"
"腹" (fù)  ("stomach") "B-DISE"
"瀉" (xiè)  ("pour") "I-DISE"
"!" ("!")
```

The test data of the *CHNER* datasets has two editions, “test file” and “truth file”. “test file” contains only the text data. Each character is saved as a singular value on a separate line. Meanwhile, the “truth file” records both the character with the gold label on a new line, divided by a white-space. Empty lines are used to distinguish different sentences. Even though there are word boundaries suggested for the training data, the test data of the shared task focused only on character based prediction.

Participants of the shared task are free to utilize additional resources within three attempts. The models are required to predict CHNER labels for the “test file.” The predicted labels are compared with the gold labels from the “truth file” for evaluation of the model performance.

If the text is a sentence of 10 Chinese characters with gold labels. The model is expected to predict 10 corresponding labels for comparison with the gold labels. However, the length of different predictive models could vary based on different segmentation schemes of the training samples. For example,

```
Text:
"心臟很痛是體力下降？ "
(Xīn zàng hěn tòng shì tǐ lìxià jiàng ?)
("Is severe heart pain a sign of physical decline?")

Gold labels:
"B-BODY","I-BODY","O","O","O","B-SYMP","I-SYMP", "I-SYMP","I-SYMP", "O"

Input 1:
"心" "臟" "很" "痛" "是" "體" "力" "下" "降" "？ "
(Xīn zàng hěn tòng shì tǐ lì xià jiàng?)
("heart", "guts", "sever", "pain", "is", "body", "strength",
"down", "decline", "?")
```

```
Output 1:
"B-BODY","I-BODY","O","O","O","B-SYMP","I-SYMP", "I-SYMP","I-SYMP", "O"

Input 2: "心臟" "很痛" "是" "體力下降" "? "
(Xīn zàng hěn tòng shì tǐ lì xià jiàng?)
("heart", "sever pain", "is", "physical strength", "decline", "?")

Output 2:
"B-BODY","O","O","B-SYMP", "O"
```

The predictive model should output labels that match the input text. However, not all input align with the gold labels in length (e.g., input2). The mismatch between predicted labels and gold labels require further processing. We will cover this part in chapter 3.

ROCLING 2022 Shared Task[2] adopted a strict evaluation framework (exact-match evaluation) (Li et al., 2020) that confines the correct predictions to matches (of prediction and annotation) in both span (i.e., “IOB” notation) and entity type (e.g., “SYMP”). Attempts that meet the requirements are counted as correct predictions. The standard NER evaluation metrics (precision, recall and f1) were used by the shared task. The performance is judged by the macro averaged F1-score of all classes to reflect the core capabilities of the model comprehensively.

Among the seven participating teams, a hybrid model (W2NER) (Ma et al., 2022b) ranked top in the shared task. Ma et al. (2022b) adopted a multi-layer architecture presented in figure 2.2, which consists of: (1) Encoder layer; (2) Convolution layer; (3) Co-predictor layer. In the encoder layer, the input text was transformed into word vectors by BERT and BiLSTM model and passed on to the convolution layer. The Conditional Layer Normalization (CLN) is applied to obtain distance, word, and region embeddings. Subsequently, the embeddings are processed through dilated convolution and pushed forward to the Co-predictor layer, where a biaffine predictor and a multi-layer perceptron (MLP) predictor were used to generate the matrix representation that encodes the relationships between characters. Their best results reached a 0.819 F1 score on the test data.

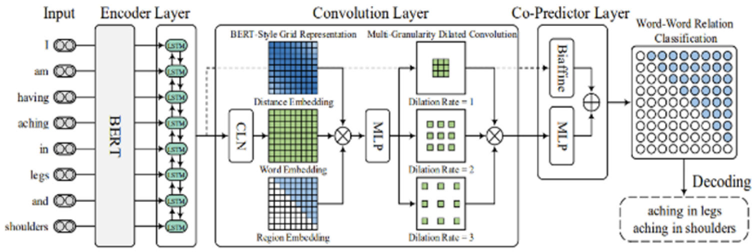


Figure 2.2: W2NER architecture (picture borrowed from Ma et al. (2022a))

Other participants from the leaderboard are listed in table 2.3. We see many teams utilized BERT (Devlin et al., 2018) for building the language models and achieved promising results, for example, SCU-MESCLab (Yang et al., 2022) deployed RoBERTa for generating embeddings of the sentences, which was later used as input of the BiLSTM-CRF standard deep learning architecture. NERVE (Lin et al., 2022) team

[2] https://github.com/NCUEE-NLPLab/ROCLING-2022-ST-CHNER

utilized BERT transformers and a lexicon-based model and achieved the best F1 score of 0.7569. The crowNER (Zhang et al., 2022) team carried out adversarial learning and mixed precision training methods to enhance the performance of the MacBERT (Cui et al., 2021) based model and obtained the best result (0.807 f1) with the MacBERT-CRF system. NCU1415 (Feng et al., 2022) experimented using BERT-based models (e.g., RoBERTa (Liu et al., 2019) and BART (Lewis et al., 2019)) for sentence encoding, followed by sequence labeling of the CRF classifier. Their best model achieved a f1-score of 0.726. YNU HPCC (Luo et al., 2022) employed focal loss and regularized dropout mechanisms to improve the BERT-BiLSTM-CRF model. Their best model ranked a fourth place in the shared task with the F1-score of 0.7768.

More traditional algorithms such as Random forest, HMM, CRF were explored by the SCU-NLP lab (Yang et al., 2022). They compared the generalization of the models with BERT, and conducted error assessments on these models.

Overall, BiLSTM-CRF networks emerged as the most applied neural architecture of the shared task, consistently yielding promising results.

Team	Precision	Recall	F1	Keywords
MIGBaseline	0.819	0.818	0.819	W2NER
SCU-MESCLab	0.801	0.783	0.792	BiLSTM-CRF (Roberta)
crowNER	0.778	0.781	0.779	MacBERT-CRF
YNUHPCC	0.772	0.781	0.776	BERT-BiLSTM-CRF
NERVE	0.796	0.730	0.762	3 NER frameworks (BERT, SoftMax)
NCU1415	0.746	0.728	0.737	BERT-CRF
SCU-NLP	0.647	0.779	0.707	Random Forest, HMM, CRF, BERT

Table 2.3: Participants of ROCLING 2022 Shared Task

Despite most systems achieving fair performance on mining the healthcare entities, we still see room for improvement. Especially on the study of word segmentation in relation to NER performance, only a few teams discussed the topic with traceable experimental records. The wining team (Ma et al., 2022b) explored using word-level input instead of characters. They witnessed a performance drop after switching the original word-based data with self-implemented word-based data. They were convinced the degrade of word-cut accuracy triggered the situation. Their questions remained unsolved: what type of word segmentation enhances NER performance? (bigger word? medium word? smaller word?)

SCU-NLP (Chiou et al., 2022) questioned the effectiveness of using characters as input for extraction of healthcare entities. They speculated that using characters as input is doomed by the lack of word boundaries based on the discovery that character-based models made more position errors than word-based systems, especially in mixing the “B” and “I” notations.

Motivated by such research topic, we were determined to find out evidence that support their hypothesis.

2.4 Chinese Word Segmentation (CWS)

As the intermediate step of many Asian Language Processing (ALP) tasks, **word segmentation (WS)** plays important role in providing word boundary information, which is preliminary for more complex language processing, such as part-of-speech (POS) tagging and parsing, text classification (TC) and Machine Translation (MT).

2.4.1 Challenges

Never far from controversy, the question of "what is a Chinese word?" has been discussed for years. Unfortunately, there is no simple solution, primarily due to lack of delimiters such as empty space or strong external cues such as inflection and capitalization.

These signals are crucial for identifying individual semantic parts in English and many other western languages, e.g., "吃雞胸" (chī jī xiōng) ("eating chicken breast") has no explicit meaning separation cues, except for the punctuation mark at the end. Another linguistic characteristics of Chinese is semantic ambiguity. For example, "雞胸肉" (jī xiōng ròu) ("chicken breast") can be simply the food, it can also be split into "雞胸" (jī xiōng) ("chicken breast") and "肉" (ròu) ("meat"), or "雞" (jī) ("chicken"), "胸" (xiōng) ("breast"), and "肉" (ròu) ("meat"). The splitting of words changes the semantic meaning of the text within limited context, not to mention text with more characters. Moreover, the migration of words has introduced many modern expressions in Chinese. For instance, "盤" (pán) ("disk") emerges very frequently on Chinese social media (e.g., Weibo). It has the more up-to-date meaning of "playing around." Faced with the challenges, segmenting Chinese text is not as intuitive as many western languages such as English.

2.4.2 Methods

Lin et al. (2020) summarizes CWS methods into two branches: traditional machine learning approach and deep learning methods. Traditional machine learning models rely on word features for realization of word tagging. Various features were implemented for the task, such as sub-token information, which refers to the use of smaller linguistic units, such as morphemes or character sequences that potentially helps the model capture more morphological and semantic characteristics of the input. An example of traditional ML model is semi-Conditional Random Fields (semi-CRF). However, the methods have limitation in training efficiency and generalization ability.

Deep learning methods change the situation by deploying word embeddings and neural network structures for extracting more upscale syntactic and semantic information from the input characters or words. More recently, deep learning models with simpler network structures have overcome the problem of high computational cost through utilizing mixed inputs (character-word) accompanied by more effective training strategies to achieve performance and efficiency close to the traditional methods.

Many recent research split the problem into two sub-tasks, i.e., word boundary disambiguation and unknown word identification (Fu et al., 2008). Since handling out-of-vocabulary (OOV) words and ambiguities are equally important for addressing the issue.

On the one hand, setting an appropriate set of rules to deal with unseen words is essential for distinguishing word borders, e.g., a more tolerant CWS system will pass on "上海復旦大學" (shàng hǎi Fù dàn Dà xué) ("Shanghai Fudan university") as one entity without knowing the word "復旦" (Fù dàn) ("Fudan"). Meanwhile, a less tolerant system breaks the sequence into more than one pieces, despite witnessing "上海" (Shàng hǎi) ("Shanghai") and "大學" (Dà xué ("university") occurring multiple times. On the other hand, assigning a character to the right belonging is crucial for dealing with unknown words, e.g., a more sensitive system assigns, “二” (èr) (“two”) to “第” (dì) (“number...”) instead of “手” (shǒu) (“hand”) when the model saw “第二” (dì èr) (“second”) co-occurring more often than “二手” (èr shǒu) (“second-hand”), despite the absence of the third character in the dictionary. The fairness of a word-cut system hinges on both sub-tasks, hence the two pointers are intertwined.

The evaluation of CWS performance is beyond the scope of this experiment. But in principle, dictionary-based CWS tools tend to output text segmented as longer fragments with more domain lexicon provided, e.g., “血清胺基丙酮酸轉化酶” (xuè qīng ān jī bǐng tóu suān zhuǎn huà méi) (“Serum Glutamic Pyruvic Transaminase”) is split into multiple parts when the vocabulary is absent in the dictionary for segmentation.

2.4.3 Jieba

“Jieba” (Chinese for “stutter”) is a python library[3] that specializes in Chinese text segmentation (both traditional and simplified Chinese.) The model uses a directed acyclic graph (DAG) based on dictionaries to provide efficient word graph scanning. The module also adopts a Maximum Matching (MM) algorithm that calculates the likelihood of all possible combinations of characters from sequence and selecting the most compatible pairs based on the frequency of usage. Moreover, the model includes a Hidden Markov Models (HHMs) layer and “Viterbi” algorithm to help the decision making of word boundaries, especially among unknown words. By default, the “HHMs” layer is not enabled since it may affect the segmentation accuracy in both ways.

The module mainly depends on three types of probability tables to calculated the probing of each character: (1) Transition probabilities between four states, i.e., B (beginning), M (middle), E (end), and S (single word); (2) Emitting probabilities from position to single characters; (3) The probability of a word starting with a certain state, i.e., the numeric value of (“體” |M) represents the probability of the character “體” (tǐ) (“body”) appearing in the middle of a word.

The default dictionary uses lexicon from People’s Daily’s newspaper (since 1998), and Microsoft Research (MSR) corpus. The size of the dictionary was expanded by adding literary text from online novels. Besides, two extra dictionaries are available on the official website for users to adjust the segmentation dictionary according to their research focus. They are a smaller dictionary with less memory footprint and a bigger dictionary that better supports traditional Chinese (繁體).

Jieba also allows modifying the default dictionary. Users can add new vocabulary or delete existing vocabulary in the default dictionary. Jieba supports modifying the dictionary in bulk. Users can enriched the default dictionary by loading a “UTF-8” encoded *txt* file that contains data (word) and metadata (word frequency, POS tag, but not mandatory) of the word samples. Here is a text example:

```
"凱特琳 1 nz" (Kǎi tè lín) ("Katherine")
```

We see three pieces of information: (1) the vocabulary; (2) the frequency of the vocabulary (“1”); (3) the part of speech tag of the vocabulary (“nz”, which refers to “other proper nouns”). All information is stored on the same line separated by white spaces.

The system maintains the balance between accuracy and speed by switching between different modes: Accurate Mode, Full Mode, and Search Engine Mode. We use the following example to demonstrate the segmentation difference between the modes:

[3] https://github.com/fxsjy/jieba

```
Example sentence:
 “我來到北京清華大學"
(wǒ lái dào běi jīng qīng huá dà xué)
( “I came to Beijing Tsinghua university")

Default mode:
 “我", “來到", “北京", “清华大学"
(wǒ lái dào běi jīng qīng huá dà xué")
( “I", "came to", “Beijing", “Tsinghua university")

Full mode:
"我", "來到", "北京", "清華", "清華大學",
"大學", "華大"
("wǒ", "láidào", "běijīng",  "qīnghuá", "qīnghuádàxué",
"dàxué", "huádà")
( “I", "came to", "Beijing", "Tsinghua", "Tsinghua university",
 “university", “Huada")

Search Engine mode:
 “我", “來到", “北京", “清華", “清華大學",
"北京大學", "華大", "北京清華"
(wǒ lái dào běi jīng qīng huá qīng huá dà xué
běi jīng dà xué huá dà běi jīng qīng huá)
( “I", "came to", "Beijing", "Tsinghua", "Tsinghua university",
"Beijing university", "Huada", "Beijing Tsinghua")
```

Accurate Mode (Default mode) prioritizes the word-cut accuracy, it is set as default mode, since it meets the requirement of most text analysis, the processing speed of *Default Mode* is 400 KB per second; *Full Mode* ensures the speed of processing through simply listing all possible separation results of the sequence, whose application is limited due to lack of accuracy. Text is processed at a speed of 1.5 MB per second under such mode; *Search Engine Mode* cuts the sequence into several shorter words in the attempt to increase recall rate, which is primarily adopted by search engines.

2.4.4 Other CWS tools

Apart from such as *Jieba*, there are other tools such as *HanLP*, and *PKUSEG* available for efficient Chinese word segmentation.

HanLP[4] is a comprehensive toolkit invented for different NLP tasks in multiple languages (more than 130), including Chinese (simplified and traditional), English, Japanese, Russian, French, and German (He and Choi, 2021). *HanLP* is famous for its versatility, efficiency, up-to-date corpora, transparent architecture, and customization capabilities.

HanLP utilizes models that are pre-trained on a multi-domain corpus (news, social media, finance, law) that consists of 99.7 million characters for the segmentation task. The model has two modes. The “coarse-grained” mode, as the default mode, prioritizes the accuracy of word segmentation, which is more popular in text mining. The “fine-

[4] https://github.com/hankcs/HanLP

grained” mode aims to split the text into different segmentation possibilities, which is more suitable for search engines. For example,

```
Input sentence:
"自然科技公司”
(zì rán kē jì gōng sī)
("natural science company")

Coarse-grained mode:
"自然", "科技", "公司"
(zì rán kē jì gōng sī)
("nature", "technology", "company")

Fine-grained mode:
"自然", "科技", "公司” , "自然科技", "科技公司”
(zì rán, kē jì, gōng sī, zì rán kē jì, kē jì gōng sī )
("nature", "technology", "company", "natural science", "technical company")
```

PKUSEG[5] is a versatile Chinese word segmentation toolkit Chinese word segmentation (Luo et al., 2019). In the attempt of achieving more accurate word-cut, *PKUSEG* approached the problem with a multi-domain segmentation support scheme, where several pre-trained models that specialize in different domains were utilized accordingly.

2.5 External Data Resources

Available Chinese healthcare resources are normally medical lexicons, bioinformatic corpus, and other ontology, e.g., Chinese DBpedia knowledge base (CN-DBpedia), which is a general purpose ontology that covers lexicon of various industries, including the medical field. Extracting healthcare information from CN-DBpedia is time-consuming since the corpus is nested in a hierarchical way. Besides, the quality of data is not guaranteed without fact-checking, which raises concerns of deploying these resources.

We chose existing domain-specific data based on accessibility and accountability of the data. After searching for available resources, we found external corpora that contain considerably large amount of Chinese healthcare lexicon. The test data were excluded from the external corpora due to absence of NER information. Now we introduce each corpus with more details.

CBLUE[6] (Chinese Biomedical Language Understanding Evaluation) is a Chinese biomedical information processing leader board led by CHIP (China Health Information Processing) Committee (Hongying et al., 2021). The organization is founded by multiple companies and education institutions, such as Yidu Cloud Technology Inc., Tencent Jarvis Lab, Sun Yat-Sen University, etc. The founding aims to nurture the advancement of Chinese biomedical natural language processing (BioNLP) technology (Zhang et al., 2021). Information about the agenda and previous work is published on the “Tianchi” platform, public can access the resource for free after approval of application. *CBLUE* was first released at the CHIP2020 conference (Guan et al., 2020). It is the composition of several domain corpora originating from different sources, such as

[5] https://github.com/lancopku/pkuseg-python

[6] https://tianchi.aliyun.com/dataset/95414

clinical trial registration, electronic health records (EHRs), online healthcare forums, search engine logs, and textbooks. The labeling of text samples were based on the majority voting of domain experts (3 to 5 on average), the annotations reached 0.9 Fleiss' Kappa scores across different annotator pairs.

Table 2.4 provides more details about the *CBLUE* datasets regarding the data distribution, NER examples (from train, development, test sets) and corresponding tasks of the corpora. We only selected four datasets that are relevant to our experiment, **CHIP-CDN, CMeEE, CMeIE, IMCS-V2-NER.** We introduce each corpus in relation to our experiment individually.

Dataset	Task	Train	Dev	Test
CHIP-CDN	Diagnosis Normalization	6,000	2,000	10,192
CMeEE	Named Entity Recognition	1,5000	5,000	3,000
CMeIE	Information Extraction	14,339	3,585	4,482
IMCS-V2-NER	unknown	unknown	unknown	unknown

Table 2.4: Overview of task-specific corpora from the CBLUE datasets

CHIP-CDN (Clinical Diagnosis Normalization) dataset contains final diagnoses of Chinese electronic medical records collected from multiple medical departments of Class A and tertiary hospitals after filtering out the privacy information (Zhang et al., 2021). All data was collected from the Chinese Clinical Trial Registry (ChiCTR) website[7]. The dataset aims to address the chaos of co-existing terminologies with identical clinical meaning, e.g., "肺佔位性病變" (fèi zhàn wèi xìng bìng biàn) ("pulmonary occupying lesion") and "胸膜佔位" (xiōng mó zhàn wèi) ("pleural mass") are both recognized as the clinical disease of "space-occupying Lesion of the Lung". The *CHIP-CDN Dataset* was annotated by Yidu Cloud team. Several medical workers from clinical background participated the annotation. The annotations were examined by peer review and ground-truth checking. The annotated diagnosis records were saved as a list of dictionaries in the following format:

```
{
    “text": "左膝退變伴遊離體"
   (zuǒ xī tuì biàn bàn yóu lí tǐ)
   ( “left knee Degeneration accompanied by joint effusion"),

    “normalized_result":  “膝骨關節病##膝關節游離體"
   (xī gǔ guān jié bìng ## xī guān jié yóu lí tǐ)
   ( “Knee osteoarthritis ## knee joint disposition")
}
```

We see two keys in each dictionary of the list, i.e., "text" provides entry of an ICD standard vocabulary, "normalized_result" field contains the "normalization" results of the text. We simply extract the value of "text" and obtained 3849 unique entity names from the datasets.

CMeEE (Chinese Medical Named Entity Recognition Dataset) is a biomedical corpus assembled from various sources such as textbooks, encyclopedias, clinical trials,

[7] http://chictr.org.cn/

medical literature, electronic health records, and medical examination reports. The *CMeEE dataset* includes 15,000 samples in the training set, and 5,000 samples in the development set, was designed for the study of retrieving mentions of medical terminologies, such as medical procedures (pro), body (bod), medical examination items (ite), microorganisms (mic), department (dep). A team of 32 annotators with different backgrounds participated in the annotation,i.e., biomedical informatics, medical, and computer science. Two trained medical experts were in charge of creating the annotation guidelines. The final results reached a high level of IAA (0.8537 Kappa score). All data and metadata were saved as a list of dictionaries, here is an example:

```
{
    "text": “細胞減少與肺內病變。"
    (xì bāo jiǎn shǎo yǔ fèi nèi bìng biàn.)
    ( “Cellular reduction and pulmonary lesions."),

     “entities": [
        { “id": 0,  “start_idx": 0,  “end_idx": 2,  “type": bod,

         “entity":  "細胞" (xì bāo)  ("cell")}...]
...}
```

“text” is used for storing all textual data, while “entities” records a list of sub-dictionaries that contain mention(s) of healthcare name(s) along with the start and end index. In total, we extracted 40545 unique entity names from the datasets.

The **CMeIE** (Chinese Medical Information Extraction) Dataset is the sibling corpus of the *CMeEE* dataset. The **CMeIE** dataset was annotated by the same annotators of the *CMeEE* dataset, with an IAA of 0.83. The dataset is dedicated to creating a benchmark for the Chinese biomedical Entity Recognition and Relation Extraction (RE). RE involves identifying the semantic relation(s) of existing entity pair(s) in the sentence (Qin et al., 2021). The information of entities relationships is essential for more complex language processing such as building knowledge graphs (KG), natural language understanding. The corpus is saved in the following format:

```
{
     “text": “藥物治療疾病。"
    (yào wù zhì liáo jí bìng) ( “drug treatment of diseases"),

     “spo_list":[{ “subject_type” :  “疾病" (jíbìng)  ( “disease"),

     “predicate":  “藥物治療" (yào wù zhì liáo)  ( “chemical treatment"),

     “word_label": [ “O",  “DISE",  “O"],

     “object_type” :  “藥物" (yào wù) ( “chemical") }...]
}
```

each “text” instance is accompanied by a “spo_list”, which is a list of sub-dictionaries that contain the keys of “subject_type”, “predicate”, “word_label”, and “object_type”. The example shows the subject (“疾病”” (jí bìng) (“disease”)) and the object (“藥物” (yào wù) (“medicine”)) are connected by the predicate type “藥物治療” (yào wù zhì liáo) (“medical treatment”). We only extract the “subject_type” and “object_type” for our experiment due to relevance pf the CHNER task.

IMCS-V2-NER saved data and meta data as dictionaries with several keys: “diagnose”, “self-report”, “explicit-info”, “dialog”, “report”, “implicit-info”, we simply extract content from “diagnose” since it is the only key holds NER information. In the end, we obtained 431 unique entity names from the *IMCS-V2-NER* datasets.

THUOCL (THU Open Chinese Lexicon)[8] is a refined Chinese lexicon knowledge base compiled by Tsinghua University NLP Lab and Computational and Social Humanities. The corpora covers vocabularies across 11 different domains, such as IT, finance and economics, idioms, etc. The data were collected from three main sources during different time spans, see table 2.5. All lexicon from each domain were saved in a text file, with the vocabulary and the document frequency (DF) value recorded on each line. DF is normally used to imply the relative importance of a term inside of a certain document. Frequency statistics of the corpus was based on the data collected during a specific time-frame from different sources. For our experiment, we used *THUOCL_medical.txt* since it is the only file that contains domain data.

Source	Start Time	End Time	Documents
CSDN Blog	2014.07	2016.07	3,785,976
Sina News	2008.01	2016.11	8,421,097
Sogou Corpus	unknown	unknown	729,008,561

Table 2.5: THUOCL data and metadata

Chatbot (Chatbot-base-on-Knowledge-Graph)[9] is a clean dataset based on healthcare related questions and answers. The data was collected from “Xunyiwenyao”[10], which is a public healthcare consulting website for conversations between patients and doctors. The corpus was devoted to studying a series of language processing questions such as sentence parsing, knowledge graphs manufacturing, and knowledge points querying. The NER information was provided in files named after “check”, “department”, “disease”, “drug”, “food”, “producer”, “symptom”. Each file was used to recorded all instances of a predefined sub-category of healthcare entities, e.g., “producer” are full collection of current medicine names in the dataset, each entity name is recorded on a new line.

Besides, more medical domain Chinese data resources are listed in Appendix A.1, we did not choose them due to availability reason or irrelevance to the domain.

[8] https://github.com/thunlp/THUOCL/tree/master

[9] https://github.com/baiyang2464/chatbot-base-on-Knowledge-Graph

[10] https://www.xywy.com/

Chapter 3

Methodology

The experiment explores the impact of word segmentation on NER performance, our experimental set-up was based on the code[1]. The original data was provided as characters by the CHNER datasets. In addition to that, we create 3 other versions of the corpora that deploy different segmentation schemes. Through utilizing *Jieba* supported by different dictionaries (Default, ROLING, FUSION) with varying number of healthcare vocabularies, we obtain segmentation results with longer spans.

By principle, adding more task-specific vocabularies into the segmentation dictionary increase the chances of cutting the text into bigger pieces (longer spans). Especially in the healthcare domain, where many healthcare names have 2 characters or more. The adapted data and the original data are used as the training corpora using the same model design (BiLSTM-crf) for developing 4 corresponding NER classifiers. The development data is pre-processed in the same way for finding the primal configuration of each classifier. We simply carry out hyper-parameter on these systems by comparing different number of epochs (3, 5, 7.) After the training, we compare the results of the three novelty systems to find out the best classifier, the selected classifier (word-level) will be compared with the classifiers trained on the original character-based corpus to validate on the test data to explore the impact of segmenting text into longer sub-tokens on NER performance. Or to put it in simpler words, the experiment aims to find out whether changing the amount of domain lexicon of the segmentation dictionary in *Jieba* improves predictive models on the CHNER task.

Based on the method described above, we carry out the experiment by the following steps: (1) Data pre-processing; (2) Training and hyper-parameter tuning the BiLSTM-CRF classifiers; (3) Model predicting; (4) Post-processing. We discuss each step separately and provide more details.

3.1 Data Pre-processing

The data pre-processing involves: (1) Building segmentation dictionaries with different healthcare lexicons (CHNER, FUSION); (2) Using *Jieba* equipped by different segmentation dictionaries to pre-process the training and development data; (3) align the gold labels to the new segmentation boundaries.

As mentioned in section 2.3, the original training corpus provides data and metadata. We only extract healthcare lexicon from the training and development sets and

[1] https://github.com/xiaofei05/Chinese-NER

ensure the test set remains untouched. We extract data from the “character” and “character label” columns of each file. We also build up a dictionary (i. g., *CHNER* dictionary) that includes all existing healthcare entities based on available gold labels. The example below is a sentence from the original data, the characters and corresponding gold labels can be easily extracted from the ”character” and ”character-label” columns through iterating each character of the sequence with the annotations, e.g., from the sample, we discover three entities, i.e., “囊腫” (náng zhǒng) (“cyst”) is a case of “DISE” entity, “增大” (zēng dà) (“enlargement”) and “疼痛” (téng tòng) (“pain”) are two instances of “SYMP” type.

```
character:
“囊，腫，的，增，大，可，導，致，嚴，重，疼，痛 "。
(náng, zhǒng, de, zēng, dà, kě, dǎo, zhì, yán, zhòng, téng, tòng, .)
( “Cyst swelling can lead to severe pain".)

character-label:
"B-DISE, I-DISE, O, B-SYMP, I-SYMP, O, O, O, O, O, B-SYMP, I-SYMP, O"
```

(2) Apart from the *CHNER dictionary*, we also used the corpora mentioned in section 2.5 for external domain-specific vocabularies. We only use on the training and development data of the corpora since NER annotations are only available in the train and development datasets.

The raw data was filtered by: (1) Leaving out irrelevant content, for example, special characters such as “[”, “]”, “(”, “)” were discovered in mix of the domain content, therefore, we created a list of special characters to removed irrelevant content and preserve the text between special characters; (2) Setting the threshold of entity length as 12 based on the observation that longer text tend to be non-entities; (3) Merging duplicates (exact matches of vocabulary) to maintain singularity of the vocabularies. We followed the pre-processing steps mentioned above and obtained the *chatbot* dictionary and *THUOCL* dictionary; The retrieved data from *CBLUE* datasets was further processed since *CBLUE* datasets contains multiple sources of data. We simply abid the same pre-processing rules (filter irrelevance, setting threshold, merge duplicates) for combining content from different corpora (*CHIP-CDN, CMeEE, CMeIE, IMCS-V2-NER*), and obtained the *CBLUE* dictionary.

Table 3.1 shows the entities, mean length, standard deviation length, and compilation information of the external healthcare dictionaries (*THUOCL, Chatbot, CBLUE).* The *CBLUE* dictionary mounts the highest entity names of 47,415, followed by the *Chatbot* dictionary (40,102 entities), the two dictionaries together provide more than 80% of the external healthcare entities used in our experiment. In comparison, the *THUOCL* dictionary comprises 18,745 entities. Meanwhile, other statistical features of the *CBLUE* and *Chatbot* dictionaries stay on a similar level, with the mean length (6.2 and 6.9) and standard deviation (2.6 and 2.8) respectively; The measurement of the *THUOCL* dictionary remains the lowest, with the mean length (mean) and standard deviation (std) of 4.2 and 1.7.

The extracted external dictionaries (*THUOCL, Chatbot, CBLUE)* were joined together as the *SUP* dictionary, and then joined by the existing lexicon (*CHNER Char*) to create the ultimate lexicon (*FUSION*).

Table 3.2 presents the entity count, mean length and standard deviation of the entities, and compilation details of the domain dictionaries. The *CHNER dictionary* contains 68,460 entities with the mean and standard deviation of 3.7 and 1.7, in comparison, the *FUSION dictionary* are approximately 1.5 times in size (106,160) after absorbing vocabularies from both *SUP dictionary* and *CHNER dictionary*, which marks the peak point of healthcare lexicon number among all dictionaries, with the mean length and standard deviation of 6.1 and 2.8.

Datasets	Entities	Mean	Std	Compilation
THUOCL	18,745	4.2	1.7	THUOCL
Chatbot	40,102	6.9	2.8	Chatbot
CBLUE	47,415	6.2	2.6	CHIP-CDN, CMeEE, CMeIE, IMCS-V2-NER

Table 3.1: Domain Dictionaries Overview (number, mean, standard deviation of entities)

Dictionaries	Entities	Mean	Std	Compilation
CHNER	68,460	3.7	1.7	CHNER
SUP	97,376	6.3	2.7	THUOCL + Chatbot + CBLUE
FUSION	106,160	6.1	2.8	SUP + CHNER

Table 3.2: Segmentation Dictionaries Overview (number, mean, standard deviation of entities in training and dev data)

(3) We used the dictionaries generated from previous step for implementation of the *Jieba* user-defined dictionary and obtained three versions of the task datasets. We named the segmented corpora “Jieba Base”, “Jieba Upgrade”, “Jieba Full”. Table 3.3 shows the mean lengths of the segmented pieces of the three word-based and the character-based task data (training and dev datasets).

The mean lengths of the “CHNER Char” datasets are exactly 1.000 since all units are characters among the datasets; “Jieba Base” (default dictionary) segmented the input sentences into sub-tokens with the mean length of 1.571. When we combined “CHNER” dictionary with Jieba default dictionary, “Jieba Upgrade” witnessed a visible growth (from 1.571 to 1.626), compared to the “Jieba Base” datasets. This trend did not develop further for “Jieba Full” datasets, where the mean length stopped growing at a plateau of 1.627. Overall, the three novelty datasets exceeded the mean length of the “Baseline” datasets by more than 50%, which indicate “Jieba” tend to segment sentences into longer words.

Meanwhile, we see very different entity length distribution across the datasets, with the mean lengths of healthcare entities going in an opposite direction, i.e., “CHNER Char’ datasets (2.596), “Jieba Base” datasets (1.861), “Jieba Upgrade” datasets (1.052), “Jieba Full” datasets (1.040). This is because we applied a different rule for calculating the mean length of entity names, here is an example:

```
Original text:
“修復肌肉蛋白"
(xiū fù jī ròu dàn bái)
( “rebuilding muscle tissue")
```

```
CHNER Char datasets:
“修", “復", “肌", “肉", “蛋", “白"
(xiū fù jī ròu dàn bái)
( “fix", “restore", “muscle", “meat", “egg", “white")

Jieba Base datasets:
“修復", “肌肉", “蛋白"
(xiū fù jī ròu dàn bái)
( “rebuild", “muscle", “protein")

Jieba Upgrade datasets:
“修復", “肌肉蛋白"
(xiū fù jī ròu dàn bái)
( “rebuild", "muscle tissue")

Jieba Full datasets:
“修復", “肌肉蛋白"
(xiū fù jī ròu dàn bái)
( “rebuild", “muscle tissue")
```

The original text was recorded as 4 characters in the “CHNER Char” corpus, i.e., “肌” (jī) (“muscle”) , “肉” (ròu) (“meat”), “蛋” (dàn) (“egg”), “白” (bái) (“white”), hence the length (as characters) of the text sample is “4” in the “CHNER Char” corpus. Meanwhile, the text was regard as “肌肉” (jī ròu) (“muscle”) and “蛋白” (dàn bái) (“protein”) by the “Jieba Base” datasets, which is equal to the length (as sub-tokens) of “2”. Moreover, “Jieba Upgrade” and “Jieba Full” both treat the sequence as a single unit, which means the length (as a sub-token) is “1”. It is clear that the lengths of the entities change dramatically across different segmentation strategies. The mean length of the “CHNER Char” datasets (2.596) was almost halved in the “Jieba Base” datasets (1.861). The declination was further sharpened in “Jieba Upgrade” (1.052) and “Jieba Full” (1.040) datasets. The above statistics indicates *Jieba* fulfills the role of preserving named entities as sub-tokens bigger than characters.

Datasets	Entities	Sub-tokens	Segmentation Dictionary
CHNER Char	2.596	1.000	not required
Jieba Base	1.861	1.571	Default
Jieba Upgrade	1.052	1.626	Default + CHNER
Jieba Full	1.040	1.627	Default + FUSION

Table 3.3: Mean Lengths (or number of splits) of Entities and Sub-tokens across datasets (including train-dev-test data) using different segmentation strategies: (1) “Entities” show the mean split numbers (each entity as how many smaller sub-tokens) of the entities; (2) “Sub-tokens” indicate the mean length (in characters) of the entities)

(4) However, the segmented sequences no longer align with the original gold labels, therefore, annotations need to be updated for the “new” data. To be more specific, annotation of the characters needs to be transformed into labels that align with the segmented text sequence in both span and entity type. Therefore, the mapping of

current sub-token tag is determined by the “old” tag of the first character in current sub-token. We use the example below to demonstrate such mapping:

```
Original sequence:
“修復肌肉蛋白"
(xiū fù jī ròu dàn bái) ( “rebuilding muscle tissue")

Original labels:
“O", "O", "B-BODY", "I-BODY", "I-BODY", "I-BODY"

Segmented sequence:
“修復", “肌肉", “蛋白"
(xiū fù jī ròu dàn bái) ( “repair", “muscle", “protein")

Merged labels:
“O", “B-BODY", “I-BODY"
```

After the segmentation, the number of labels must be reduced (from 8 to 3 labels). Since the entity “蛋白” (dàn bái) (“protein”) was treated as a sub-token that takes “I-BODY” as the label. Based on the same merging scheme, the updated tags (“O”, “B-BODY”) also align with “修復” (xiū fù) (“repair”) and “肌肉” (jī ròu) (“muscle”).

After retrieving the segmented text with merged labels, we simply adapt the updated data and metadata (train-dev set) to the following format:

```
“治療" (zhì liáo) ("treat") "B-TREAT"
“胃病" (wèi bìng) ("stomach illness") "B-DISE"
“? "   (?)        ("?")     "O"

“拉稀" (lā xī) ("diarrhea")  "B-DISE"
“腹瀉" (fù xiè) ("diarrhea") "B-DISE"
“!"    (!)      ("!")        "O"
```

Each line records a sub-token (or character) and corresponding gold label separated by an empty space. Sentences will be separated by an empty line.

For the test data, we apply matching segmentation strategies on the text and record the data without the gold labels in the format below:

```
“治療" (zhì liáo) ("treat")
“胃病" (wèi bìng) ("stomach illness")
“? "   (?)        ("?")

“拉稀" (lā xī) ("diarrhea")
“腹瀉" (fù xiè) ("diarrhea")
“!"    (!)      ("!")
```

3.2 Training and Hyper-parameter tuning the Classifiers

Regarding the training and tuning of the models, we carry out the experiment in five steps: (1) Building the training corpus for word embeddings based on the training corpus segmented by *Jieba* using the corresponding dictionary; (2) Creating *word2id* dictionary; (3) Training the *BiLSTM-CRF* classifier with restructured datasets; (4) Hyper-parameter tuning the classifiers using different numbers of *Epoch* (3, 5, 7); (5) post-processing the model output and compare them with the character-based gold labels.

(1) We first transform the original training corpus (train-dev data) into the format below, each sentence is recorded on the same line with white space between words (or characters), and new sentence starts on a new line. The original training corpus was parsed as sequences of characters, we simply create a "txt" file that records each sequence on one line, characters are separated by a blank space as the format below [2]:

```
"明" "日" "就" "叫" "你" "悔" "不" "当" "初" "! "
(Míng rì jiù jiào nǐ huǐ bù dāng chū !)
( “bright"  “day" "just" "call" "you" "regret" "not" "as"
 “first" "!")
"太"  “短"  “而"  “已"
(tài duǎn ér yǐ)
("too" "short" "and" "only")
"我" "收" "藏" "北" "史" "料"  “中"  “的"  “要" "件"
(Wǒ shōu cáng běi shǐ liào zhōng de yào jiàn)
("I" "receive" "hide" "north" "history" "material" "in",
"of" "important" "item")
```

Then we utilize *Jieba* supported by the three dictionaries (default dictionary, ROLING, and FUSION) to prepare WEs the training corpora for generating the word2id dictionaries. Here is a *Jieba Base* version of the WEs training corpus:

```
 “明日"  “就"  “叫你"  “悔不当初" "! "
(Míng rì jiù jiào nǐ huǐ bù dāng chū !)
("tomorrow" "just" "make you" "regret not as first" "!")
 “太短"  “而已"
(tài duǎn ér yǐ)
("too short" "simply")
 “我" "收藏" "北史料" "中的" "要件"
(wǒ shōu cáng běi shǐ liào zhōng de yào jiàn)
( “I"  “store"  “Northern History material"  “in"  “essential piece")
```

Table 3.4 presents similar statistical characteristics (mean, stand deviation, maximum lengths) of the elements in each word2id corpus. The differences between the mean lengths across different datasets show that the WEs training corpora have the segmentation results that match those of the task data (training-dev-test data).

(2) In this step, we generate "word2id" dictionary for each system that maps unique words in the datasets (train-dev) to a unique integer ID. The numerical representation of the words is utilized by the neural networks for more efficient processing of the text. We set the word-count threshold as ”3” based on the choice of the original code. The threshold ensures only words that occur more than 3 times in the training corpus will qualify as an unique word in the "word2id" dictionary. This filtering scheme aims to obtain more condensed collection of the more frequent words within each corpus.

(3) We train the *BiLSTM-CRF* classifiers with the restructured datasets, this experiment choice was based on the model documented in Dong et al. (2016)'s paper.

[2] Pinyin and English translations on the second and third lines are not part of the format

Embedding	Mean	Std	Max
CHNER Char	1.00	0.00	1
Jieba Base	1.73	0.71	5
Jieba Upgrade	1.82	1.01	7
Jieba Full	1.82	1.01	7

Table 3.4: Lengths (in number of characters) across different word2id corpus (2 decimal)

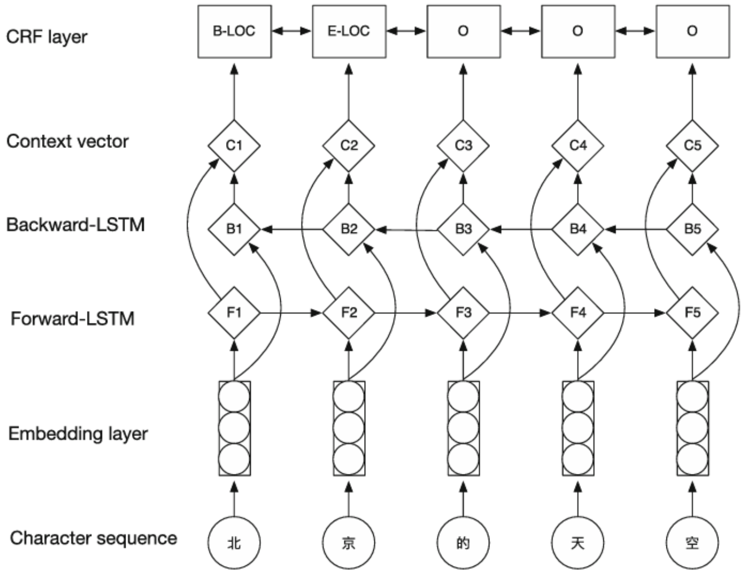


Figure 3.1: Main architecture of character-based BLSTM-CRF (picture borrowed from Dong et al. (2016))

LSTMs networks combat the long-term dependencies issue by using memory cells and gates to control the information flow. Such architecture is constructed by input gates, output gates, forget gates, and peephole connections controlled by "sigmoid" functions and weight matrices.

Figure 3.1 shows the basic mechanism of a character-level BiLSTM-CRF in capturing contextual information of a certain element in the sequence from left-to-right and right-to-left level. The vectors representations are obtained by concatenating representations from both forward and backward LSTMs. A Conditional Random Field (CRF) layer is implemented to calculate transition scores between tags and maximize the log-probability of correct tag sequences with the start and end symbols taken into account. Following the "IOB" format constraints, dependencies between output labels in the entire sentence are considered by decoding bi-gram constraints between outputs and find out the sequence with the maximum score.

The experiment was carried out in the python conda environment, using several open-source NLP tools and algorithms of specific visions for building and training the neural networks. See dependencies[3].

Besides, we refer to Wan et al. (2019)'s work to deploy parameter-tuning of the model for better generalizations, such parameters include *Epoch, Learning Rate (lr),*

[3] https://github.com/anaverageone/CHNER_using-BiLSTM-CRF.git

Batch Size, LSTM Hidden Dimension, LSTM Dropout Rate, Random Seed, GPU, Embedding Dimension. Now we present each parameter with more details.

Epoch defines how many times the entire dataset is being passed forward and backward through the NNs during the training or testing mode. Choosing the appropriate number of epochs is a trade off between model generalization and over-fitting, i.e., as the training iterations goes up, the model naturally captures more information from each looping which would no doubt help with the model prediction on certain patterns from the training corpus; However, such tendency slows down at a certain point where the learning curve is stuck at a local optimum, where the performance of the model stops improving or even decreases. Therefore, choosing an appropriate number of *Epoch* is crucial for a balanced system. For the experiment, we set the epoch as 1 to testify initial performances of the model and increase the number of *Epoch* from 3 to 5, from 5 to 7, stopping at 7.

Learning Rate (lr) decides the learning step size of the optimization algorithm at each iteration while moving toward a minimum of a loss function. It influences to what extent newly acquired information overrides old information. Selecting an adequate learning rate is pivotal, as it influences the speed and quality of model convergence. We set a *lr* of 0.01 for steady progressing towards the optimal solution during the experimentation and validation .

Batch Size represents the number of samples used in one forward and backward pass through the network and has a direct impact on the accuracy and computational efficiency of the training process. It is an important factor in balancing computational efficiency and model stability. A batch size of 256 is chosen based on the computational tractability of our feasible hardware resources.

LSTM Hidden Dimension determines the number of nodes in each hidden state of the LSTM layer. It controls the capacity and expressiveness of the model, influencing its ability to capture complex temporal dependencies. A hidden dimension of 1024 is selected to provide the model with ample capacity to learn intricate patterns in the sequential data while avoiding over-fitting.

Embedding Dimension determines the size of the embedding vectors used for encoding input tokens into a continuous vector space. Choosing an appropriate embedding dimension is essential to capture the semantic richness of the input vocabulary. 300 dimensions is assigned to training on the sub-token level input, while 200 is set as the dimension for character level training. This choice was to maintain a balance between capturing intricate semantic nuances and maintaining computational efficiency. This choice was based on the settings from the sample code.

Dropout is a regularization technique that specifies the probability of dropping out a node in the networks by randomly setting a fraction of input units to zero during training. It is widely applied in deep learning to prevent over-fitting, where a complex model loses generalization ability on future unseen cases due to rigorously fitting the model on training samples. For such reason, applying a reasonable *Dropout Rate* to the recurrent connections within an LSTM cell the training of the model on large amounts of data. A *Dropout Rate* of 0.1 is employed in this study to introduce moderate regularization, mitigating the risk of over-fitting while retaining model expressiveness.

Random Seed parameter essentially initializes the pseudo-random number generator that ensures that repeating same the training action with all conditions intact triggers identical results. We set a *Random Seed* value of “2020” facilitates the reproducibility of the experiments and fosters transparency and accountability in the research process.

The input of the model was structured in the following format, here is an example from the *CHNER* datasets:

```
“油" (yóu) ("oil") "B-CHEM"
“脂" (zhī) ("fat") "I-CHEM"
“應" (yīng) ("should")"O"
“避" (bì) ("avoid") "O"
“開" (kāi) ("open") "O"
“。 " (。 ) (".") "O"

“手" (shǒu) ("hand") "B-BODY"
“肘" (zhǒu) ("elbow") "I-BODY"
“橫" (héng) ("horizontal") "B-BODY"
“紋" (wén) ("wrinkle") "I-BODY"
“肌" (jī) ("muscle") "I-BODY"
“靠" (kào) ("lean") "O"
“近" (jìn) ("near") "O"
“身" (shēn) ("body") "B-BODY"
“體" (tǐ) ("body") "I-BODY"
“。 " (。 ) (".") "O"
```

Here is the same sentences from the *Jieba Base* datasets:

```
“油脂" (yóu zhī) ("oil") "B-CHEM"
“應" (yīng) ("should") "O"
“避開" (bì kāi) ("avoid") "O"
“。 " (.) (".") "O"

“手肘" (shǒu zhǒu) ("elbow") "B-BODY"
“橫紋肌" (héng wén jī) ("striated muscle") "B-BODY"
“靠近" (kào jìn) ("approaching") "O"
“身體" (shēn tǐ) ("body") "B-BODY"
“。 " (.) (period) "O"
```

Here is a *Jieba Upgrade* version of the input sentences:

```
“油脂" (yóu zhī) ("oil") "B-CHEM"
“應" (yīng) ("should") "O"
“避開" (bì kāi) ("avoid") "O"
“。 " (.)   (".") "O"

“手肘橫紋肌" (shǒu zhǒu héng wén jī)("brachioradialis muscle") "B-BODY"
“靠近" (kào jìn) ("close to") "O"
“身體" (shēn tǐ) ("body") "B-BODY"
"。 " (.)   (".") "O"
```

We can see the same input sentences are passed on to the BiLSTM-CRF model differently, which means the classifiers “observe” different training samples. Therefore, the preserved classifiers will no doubt generalize differently on the unseen data.

(4) Subsequently, the hyper-parameters tuning of our experiment primarily focuses on finding out the best number of *Epoch* from the pre-defined values (3, 5, 7), the

classifier with the best performance on the development data will be validated against the test set.

(5) Finally, we obtain the model output. However, the outcome of the predictive models can not be compared to gold labels directly since the annotation was provided on the character level. Models that were trained with word level segmentation need further process to convert the word-level predictions into character-level predictions (which is what is expected by the shared task.)

```
original Text:
 “油", "脂", "應", "避", "開", "。"
(yóu zhī yīng bì kāi)  ("Oil", "oil", "should", "avoid", "open", ".")

Input Sequence:
 “油脂", "應", "避開", "。"
(yóu zhī yīng bì kāi)  ("Oil", "should", "avoid", ".")

Model Prediction:
 “B-CHEM", "O", "O", "O"

Gold Labels:
 “B-CHEM", "I-CHEM", "O", "O", "O", "O"

Restored Outcome:
 “B-CHEM", "I-CHEM", "O", "O", "O", "O"
```

Such restoration is achieved by unfolding the predictions of words that contain more than one character into character-level predictions (while maintaining the BIO schema). The sentence above shows the predicted tag (“B-CHEM”) of the sub-token “油脂” (yóu zhī) (“oil”) is restored as ‘[‘B-CHEM”, “I-CHEM”], by applying the same rule to the whole sequence, the final predicted labels are restored as [“B-CHEM”, “I-CHEM”, “O”, “O”, “O”, “O”]. Length wise, the number of predicted labels should be the same as the number of expected labels (i.e., one per character). The example above shows the restored predicted labels align with the gold labels. This label flattening scheme is also applied on the predicted labels of the test data.

Chapter 4

Results

4.1 Evaluation Metrics

We adopted the same validation framework of the *ROCLING 2022 Shared Task*. Through comparing the predicted labels and human annotations in both the IOB notation and entity type, we assess the performance of each experimental system. The standard NER evaluation metrics (precision, recall, and *F1*-score) has been regarded as the de facto evaluation and optimization formula of NER performance measurement (Van Rijsbergen and Croft, 1975). We apply the metrics for measurement of the CHNER classifiers' generalization. The computing of precision, recall, and F1-score metrics are based on the calculation of True positive (TP), False Positive (FP), True Negative (TN), False Negative (FN).

TP, FP, TN, FN are values that record different situations of the predicted label and the gold label, i.e., **TP** concerns all correctly predicted instances of a certain category; **TN** are instances where the model correctly identifies a token of other class as not belonging to a particular class; **FP** are cases of other categories being classified as a certain category, which is false classification of the actual class; **FN** indicates sample of a certain category predicted as other entity types. Table 4.1 provides the matrix of *TP, FP, TN, FN*, the columns are used to represent predicted classes, while the rows record the actual belonging of the instance's category. For example, if the model correctly predicts the "B-BODY" instance as "B-BODY", it is counted as a TP; if the model yields "B-BODY" for an input belonging to the "I-BODY" class, the prediction is counted as a *FP* case of the "B-BODY" class. Meanwhile, if a real instance of "B-BODY" is identified as any non-"B-BODY" class (including "O"), the contradiction is regarded as a *FN*. Moreover, if an instance gold label is not part of this class and is classified as anything other than "B-BODY" , then it's a *TN*.

	Predicted Positive	Predicted Negative
Actual Positive	True Positives (TP)	False Negatives (FN)
Actual Negative	False Positives (FP)	True Negatives (TN)

Table 4.1: Confusion matrix (TP, FN, FP, TN)

These terms communicate confusion matrix results, which are also used in classification metrics like precision, recall, and F1 score.

Precision represent the accuracy of the model's output on each specific NE class.

The precision of each category is obtained by calculating all correctly predicted instances *TP* out of all positive predictions of the tag (*TP* plus *FP*). It measures that when a prediction for a specific class is made, how likely is that it is a correct prediction. A higher precision score means the model makes fewer mistakes in predicting instances of the given class.

Recall, also known as sensitivity, quantifies the ability to identify all existing instances of a certain class. The score equals to the number of all correct *TP* predictions divided by the summation of all true samples of the class within the dataset (*TP* plus *FN*). It measures how likely a model is to predict all instances of a class.

There is a balance between precision and recall since improving one metric often comes at the cost of the other. On the one hand, if a model focuses too much on precision, it may become overly conservative and only predict a positive result when it is highly confident, resulting in many TP getting missed out (FN increases); Vice versa, when a model prioritizes recall, it may predict positive for a larger number of instances, including many FP along with TP (FP increases).

Finding a balance between precision and recall requires taking the constraints of the task into account. For tasks such as medical diagnosis, high recall might be more important to ensure that no relevant cases are left out, even if it means a higher FP rate. In other cases, such as fraud detection or legal applications, high precision is often prioritized to minimize FP, despite at the risk of introducing more FN. For CHNER, the two metrics are equally important since we expect the language system to be cautious when predicting both the FP and FN.

F1-score is the harmonic mean of precision and recall that facilitates a more balanced performance assessment Van Rijsbergen and Croft (1975). A high F1-score suggests that the model achieves both high precision and recall for the given class.

Finally, the *macro-averaged* or *Weighted averaged* values of the evaluation metrics will be used for validation. The **macro-averaged** score is represented by the unweighted mean values of each metrics for each class. Such formula provides a comprehensive way of rating the system performance, especially when data distribution between the classes is imbalanced; The *weighted-averaged* score aggregates the performance of individual classes based on their class distribution. The algorithm assigns a different weight to each value based on the support number. The more instances a value has, the more it contributes to the final result. This approach acknowledges the varying prevalence of classes in the dataset, which ensures the computation of an overall performance of the classifier while appropriately prioritizing the influence of majority classes. However, it should be used judiciously as it may obscure the performance of minority classes if they are poorly predicted.

For more readable visualization of the evaluation results, we utilized the “classification report” and “confusion matrix” functions from the “scikit-learn” library for evaluating and diagnosing the performance of classification models.

The “classification report” function provides a summary of the evaluation metrics. In the report, the precision, recall, and F1-score of each class are examined. Users can identify areas where the model excels and areas where it may struggle, thereby guiding further model refinement and optimization efforts. Moreover, the inclusion of “support” values in the report provides information about the distribution of instances across different classes, offering additional context for interpreting the model’s performance. Support concerns only the evaluation stage (although it is likely that the same trend happens during training).

The “confusion matrix” function offers a visualization of the model’s performance by tabulating the number of correct and incorrect predictions for each class. This simplifies the locating of misclassified instances that leads to analysis of potential causes of errors.

4.2 Character-based system versus Word-based systems

As mentioned before, our designed experiment focus on finding out the two representative models (word-based and character-based) that could be utilized for the final validation on the test data. We established four experimental systems using four versions of the training data and carried out parameter-tuning by testing different numbers of *Epoch*.

Table 4.2 presents the Precision (P), Recall (R), and F1-score (F1) metrics of the four experimental sets across different Epochs.

We see the *Baseline* model they all perform quite similarly. The F1-score difference (between Epoch= 3, 5, 7) is close to 6%. The best F1 (0.676) was achieved when the Epoch number was set as 3 initially.

For the *Jieba Base* system, the best results is seen when the Epochs number equals 5. The model achieved a F1 measure of 0.624, a precision of 0.704 and a recall of 0.669 –which is a slight improvement over the character-based model. The decrease in all scores for the Baseline model trained for 7 epochs shows that the model was likely already overfitting to the training data under the parameter setting.

The *Jieba Upgrade* system presents better performance for 5 epochs, yielding the F1-score of 0.624, a precision of 0.736 and a recall of 0.555. What is interesting about this model is that it shows higher precision and F1 but lower recall when compared to the *Jieba Base* system.

The *Jieba Full* system achieved the best f1 score of 0.643 for 5 epochs, with a precision of 0.757 and a recall of 0.570. The improvement of both precision and recall compared to other Jieba supported systems (Jieba Base and Jieba Upgrade) indicate the finer generalization of the model in CHNER task for 5 epochs.

Overall, the character-based model (Baseline) shows high performance consistently, while the word-based systems (Jieba Base, Jieba Upgrade, Jieba Full) demonstrated potential in NER improvement, despite the slight lower initial performance for epochs 3, the models quickly catches on for 5 epochs 5, but the f1 measurement stopped growing when the epoch number reaches 7.

Based on the scores, we selected the Baseline (character-based, Epoch = 3) and the Jieba Full (word-based, Epoch = 5) as the two systems for evaluation on the test data.

Table 4.3 presents the classification report of the *Baseline (character-based, Epoch = 3)* system trained on the *CHNER Char* dataset. Overall, this system achieves a promising performance on the test data, with the F1-score of 0.612, a precision of 0.831 and a recall of 0.521, despite the varying effectiveness across different classes.

Table 4.4 provides insight of the Jieba Full (word-based, Epoch = 5) in predicting the healthcare entity labels of the test data, word-based system exhibit a moderate performance, yielding a F1-score of 0.642, a precision of 0.757 and a recall of 0.570.

Table 4.5 compares the macro-averaged scores (P, R, F1) of the two selected systems on the test data. We see the *Baseline (Epoch = 3)* classifier (character-based) a more competitive generalization over the Jieba Full (word-based, Epoch = 5) classifier on the CHNER task.

System	Segmentation	Epoch 3	Epoch 5	Epoch 7
Baseline (character-based)	Original Corpus	P: 0.748 R: 0.626 F1: **0.676**	P: 0.829 R: 0.519 F1: 0.610	P: 0.791 R: 0.523 F1: 0.611
Jieba Base (word-based)	Jieba (default)	P: 0.638 R: 0.361 F1: 0.441	P: 0.704 R: 0.569 F1: 0.624	P: 0.674 R: 0.495 F1: 0.561
Jieba Upgrade (word-based)	Jieba (ROLING)	P: 0.658 R: 0.300 F1: 0.378	P: 0.736 R: 0.555 F1: 0.624	P: 0.640 R: 0.471 F1: 0.536
Jieba Full (word-based)	Jieba (FUSION)	P: 0.711 R: 0.290 F1: 0.369	P:0.757 R: 0.570 F1: **0.643**	P: 0.639 R: 0.477 F1: 0.540

Table 4.2: Results on Development data (BiLSTM-crf model, Epoch “3”, “5”, “7”, “segmentation” shows the participation of jieba and the supporting dictionaries)

Classes	precision	recall	f1-score	support
O	0.866	0.984	0.922	77019
I-DISE	0.917	0.728	0.812	7571
I-BODY	0.826	0.721	0.770	8254
B-DISE	0.851	0.676	0.754	2609
B-BODY	0.776	0.716	0.745	5315
I-SUPP	0.790	0.677	0.729	551
I-CHEM	0.856	0.617	0.717	3851
B-SUPP	0.756	0.661	0.706	183
I-DRUG	0.897	0.532	0.667	1599
B-CHEM	0.789	0.544	0.644	1718
I-TREAT	0.936	0.490	0.643	1251
B-SYMP	0.774	0.532	0.630	1944
B-TREAT	0.808	0.476	0.599	468
I-EXAM	0.857	0.458	0.597	733
B-DRUG	0.811	0.455	0.583	481
I-SYMP	0.835	0.446	0.582	2878
B-EXAM	0.595	0.440	0.506	207
B-TIME	0.908	0.350	0.505	197
I-TIME	0.901	0.267	0.412	408
B-INST	0.821	0.092	0.165	250
I-INST	0.883	0.084	0.154	629
accuracy			0.860	118116
macro avg	0.831	0.521	**0.612**	118116
weighted avg	0.859	0.860	0.847	118116

Table 4.3: Classification Report of Baseline on Test data (character-based, Epoch “3”)

Classes	precision	recall	f1-score	support
O	0.816	0.986	0.893	77019
B-BODY	0.773	0.682	0.724	5315
I-SUPP	0.741	0.673	0.705	551
I-BODY	0.777	0.606	0.681	8254
B-SUPP	0.676	0.661	0.669	183
B-SYMP	0.771	0.562	0.650	1944
B-CHEM	0.829	0.497	0.621	1718
I-SYMP	0.810	0.485	0.607	2878
B-TIME	0.827	0.462	0.593	197
I-CHEM	0.892	0.442	0.591	3851
B-DISE	0.787	0.470	0.589	2609
I-DISE	0.897	0.393	0.546	7571
B-TREAT	0.773	0.429	0.552	468
I-TIME	0.881	0.380	0.531	408
I-EXAM	0.874	0.370	0.520	733
B-EXAM	0.622	0.430	0.509	207
I-TREAT	0.868	0.304	0.450	1251
B-DRUG	0.730	0.304	0.429	481
I-DRUG	0.872	0.246	0.384	1599
B-INST	0.716	0.212	0.327	250
I-INST	0.823	0.148	0.251	629
accuracy			0.814	118116
macro avg	0.798	0.464	**0.563**	118116
weighted avg	0.818	0.814	0.792	118116

Table 4.4: Classification Report of Jieba Full on Test data (word-based, Epoch “5”)

Model	Precision	Recall	F1 score
Baseline (e3)	0.831	0.521	**0.612**
Jieba Full (e5)	0.798	0.464	0.563

Table 4.5: Evaluation on Test data (character-based and word-based systems)

Chapter 5

Error Analysis

The error analysis aims to delve deeper into the errors of each model (Baseline and Jieba Full) validated on the test data. We primarily investigate the FP and FN of the models to explore the advantage(s) and disadvantage(s) of using different segmentation strategies for the CHNER task.

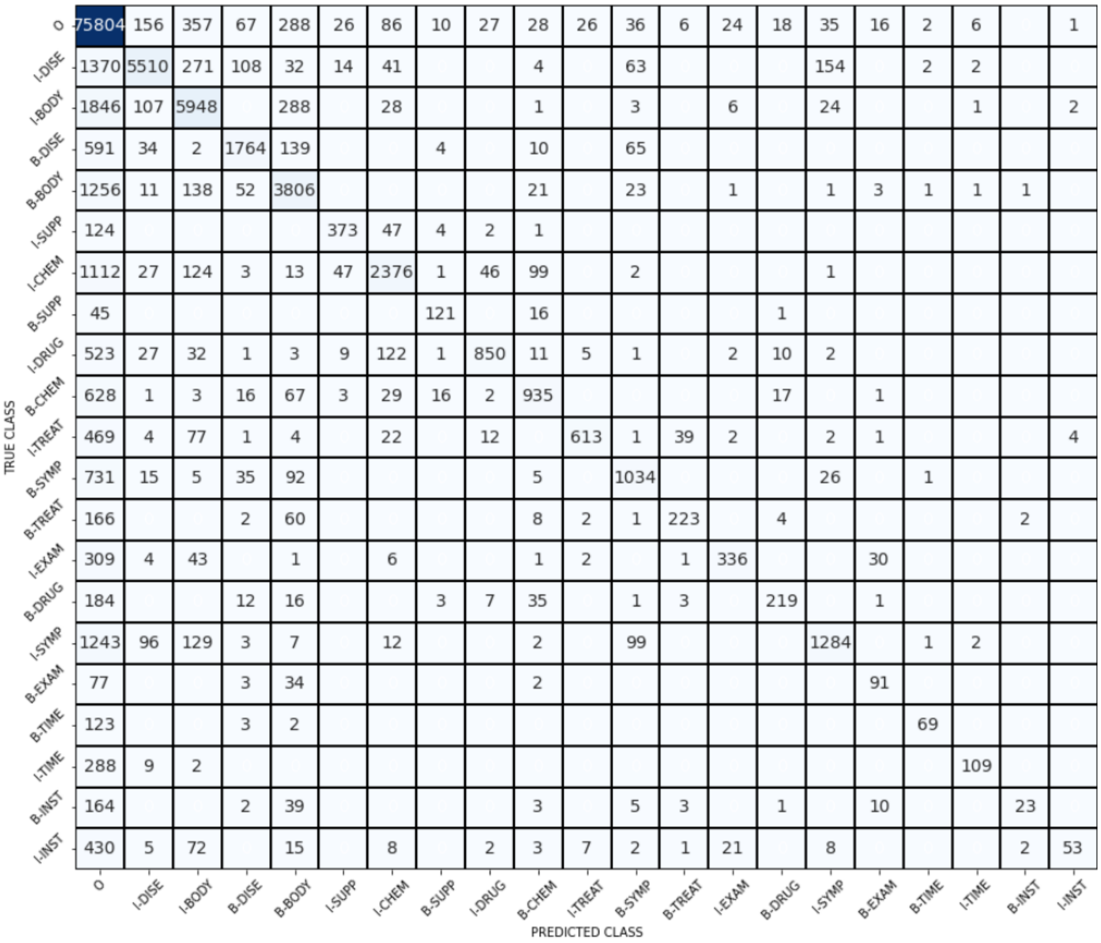
We convert the confusion matrices introduced in chapter 4 into a heatmap of the matrix with “counts” and a heatmap of the “normalized” matrix with decimal percentages.

Heatmaps offer better visualization of the model prediction count(s) compared to the regular matrix counts. Heatmaps not only sum up the four categories (TP, FP, TN, FN) of each predictive class, but also distinguishes different counts (or percentage) with shade(s) for plotting. Slots with darker shades indicate more supporting instances (or higher percentage) in the matrix. Such feature enables locating outstanding predictive behavior (the most misclassified class(es) of the predictive category) more efficiently.

In the heatmap of “counts” matrix, each cell represents the count of a specific combination of the predicted class and the real class, for example, “I-BODY” (real class) misclassified as “B-BODY” (predicted class). It’s straightforward and easy to understand. However, the “counts” matrix does not reflect the percentage of the prediction against the real class, for example, what percentage of the “I-BODY” (real class) was misclassified as “B-BODY” (predicted class). Therefore, we also use the “normalized” matrix for more convenient analysis across the predicted classes against a certain gold class. Normalizing a matrix involves converting the raw counts into percentages or proportions, often in decimal form, to make the data more comparable across rows or columns. Typically, normalization is done row-wise or column-wise. We applied the row-wise normalization, which calculates the percentage of each cell with respect to the total count in its row of the real class.

5.1 Baseline (character-based, Epoch = 3) on test data

Details about the Baseline (character-based, Epoch ”3”) system’s incorrect predictions was presented in figure 5.1 and figure 5.2. Based on our observations, our error analysis mainly focus on the relation between two groups of predictive classes: “DISE-BODY-SYMP” and “DRUG-CHEM-SUPP”. We present classification errors, i.e., false positives (FP) and false negatives (FN) of the gold labels comprehensively, overlooking the dominant class “O”. Even though this class has the largest number of instances and the highest overall score. Now we provide more details on the two groups respectively.



	O	I-DISE	I-BODY	B-DISE	B-BODY	I-SUPP	I-CHEM	B-SUPP	I-DRUG	B-CHEM	I-TREAT	B-SYMP	B-TREAT	I-EXAM	B-DRUG	I-SYMP	B-EXAM	B-TIME	I-TIME	B-INST	I-INST
O	75804	156	357	67	288	26	86	10	27	28	26	36	6	24	18	35	16	2	6	0	1
I-DISE	1370	5510	271	108	32	14	41	0	0	4	0	63	0	0	0	154	0	2	2	0	0
I-BODY	1846	107	5948	0	288	0	28	0	0	1	0	3	0	6	0	24	0	0	1	0	2
B-DISE	591	34	2	1764	139	0	0	4	0	10	0	65	0	0	0	0	0	0	0	0	0
B-BODY	1256	11	138	52	3806	0	0	0	0	21	0	23	0	1	0	1	3	1	1	1	0
I-SUPP	124	0	0	0	0	373	47	4	2	1	0	0	0	0	0	0	0	0	0	0	0
I-CHEM	1112	27	124	3	13	47	2376	1	46	99	0	2	0	0	0	1	0	0	0	0	0
B-SUPP	45	0	0	0	0	0	0	121	0	16	0	0	0	0	1	0	0	0	0	0	0
I-DRUG	523	27	32	1	3	9	122	1	850	11	5	1	0	2	10	2	0	0	0	0	0
B-CHEM	628	1	3	16	67	3	29	16	2	935	0	0	0	0	17	0	1	0	0	0	0
I-TREAT	469	4	77	1	4	0	22	0	12	0	613	1	39	2	0	2	1	0	0	0	4
B-SYMP	731	15	5	35	92	0	0	0	0	5	0	1034	0	0	0	26	0	1	0	0	0
B-TREAT	166	0	0	2	60	0	0	0	0	8	2	1	223	0	4	0	0	0	0	2	0
I-EXAM	309	4	43	0	1	0	6	0	0	1	2	0	1	336	0	0	30	0	0	0	0
B-DRUG	184	0	0	12	16	0	0	3	7	35	0	1	3	0	219	0	1	0	0	0	0
I-SYMP	1243	96	129	3	7	0	12	0	0	2	0	99	0	0	0	1284	0	1	2	0	0
B-EXAM	77	0	0	3	34	0	0	0	0	2	0	0	0	0	0	0	91	0	0	0	0
B-TIME	123	0	0	3	2	0	0	0	0	0	0	0	0	0	0	0	0	69	0	0	0
I-TIME	288	9	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	109	0	0
B-INST	164	0	0	2	39	0	0	0	0	3	0	5	3	0	1	0	10	0	0	23	0
I-INST	430	5	72	0	15	0	8	0	2	3	7	2	1	21	0	8	0	0	0	2	53

Figure 5.1: Heatmap of the Baseline system on test data (character-based, Epoch=3)

5.1.1 DISE-BODY-SYMP

`(1) "I-DISE" and "B-DISE"`

The “I-DISE” (7571 support) class has the second highest f1 score of 0.812, a high precision of 0.917 and a fair recall of 0.728. The misclassification mainly involves FP and FN of the “I-BODY”, (107 and 271 instances) As an example,

```
"頭" (tóu) ("head") B-DISE B-BODY
"皮" (pí) ("skin") I-DISE I-BODY
"屑" (xiè) ("particle") I-DISE I-BODY
```

The example shows the content in the final outcome file, where each character with the gold label followed by the predicted label are recorded on the same line, separated by white spaces. Future example follows the same format above. “皮” (pí) (“skin”) is a FN example of the “I-DISE” class misclassified as “I-BODY”. We see the three consecutive characters “頭” (tóu) (“head”), “皮” (pí) (“skin”), “屑”(xiè) (“particle”) belonging to “DISE” regarded as “BODY”. According to figure 5.2, misclassification of “I-DISE” is

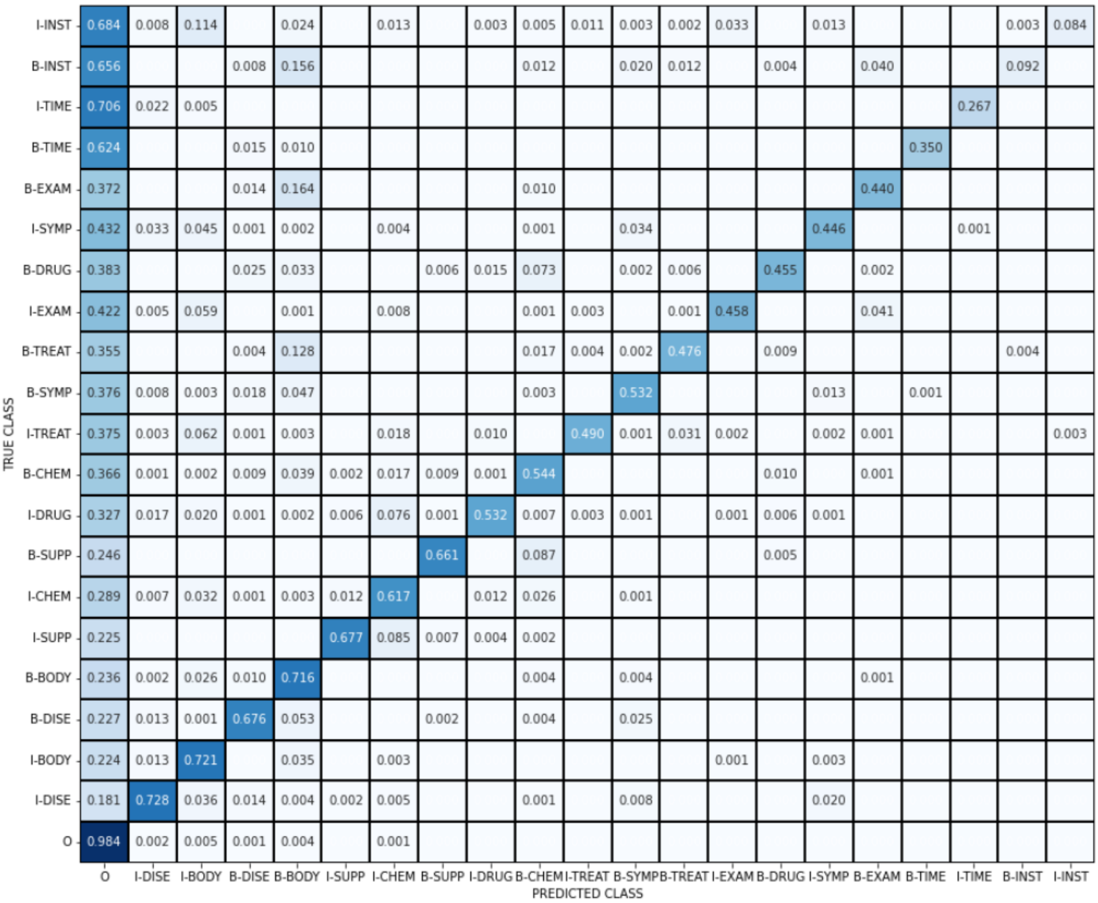


Figure 5.2: Normalized Heatmap of the Baseline system on test data (character-based, Epoch=3)

often related to “I-BODY” (0.036 percentage), while prediction errors of “B-DISE” is most connected to “B-BODY” (0.053 percentage), regardless of the “O” class.

Meanwhile, the “B-DISE” class (2609 samples) has the f1 score of 0.754, with the precision and recall of 0.851 and 0.676. The errors of “B-DISE” mostly concentrated on “I-DISE” (108 FP instances) and “B-BODY” (139 FN samples). For example,

```
"雙" (shuāng) ("duel") B-DISE B-BODY
"眼" (yǎn) ("eye") I-DISE I-BODY
"單" (dān) ("singular") I-DISE B-DISE
"視" (shì) ("see") I-DISE I-DISE
"障" (zhàng) ("obstacle") I-DISE I-DISE
"礙" (ài) ("block") I-DISE I-DISE
```

“雙眼單視障礙” (shuāng yǎn dān shì zhàng’ài) (“binocular vision impairment”) is a “DISE” entity seen as two entities by the Baseline system, i.e., “雙眼” (shuāng yǎn) (“two eyes”) (“BODY”) and “單視障礙” (dān shì zhàng’ài) (“singular vision disability”) (“DISE”.)

The confusion between “I-DISE” and “B-DISE” is most likely a word ambiguity problem, since “單視障礙” (dān shì zhàng’ài) (“singular vision disability”) already expresses “singular-eyed vision”, adding “雙眼”(shuāng yǎn) (“two eyes”) does not provide more information to the context. It is possible the Baseline system considered “雙眼” (shuāng yǎn) (“two eyes”) as an independent word, hence the misclassification.

(2) "I-BODY" and "B-BODY"

“I-BODY” (8254 cases) has the f1 score of 0.770. Misclassification of was mostly caused by “I-DISE” (107 FP) and “B-BODY” (288 FN). For example,

```
"關" (guān) ("close") O O
"脊" (jǐ) ("spin") B-BODY B-DISE
"骨" (gǔ) ("bone") I-BODY I-DISE
"神" (shén) ("god") I-BODY I-DISE
"經" (jīng) ("spirit") I-BODY I-DISE
```

The confusion between “I-BODY” and “I-DISE” (or “B-BODY” and “B-DISE”) is potentially due to training samples, where “脊骨神經” (guān jǐ gǔ shén jīng) (“spinal nerves”) is seen multiple times with disease entities such as “脊骨神經痛” (guān jǐ gǔ shén jīng tòng) (“spinal nerves pain”).

Meanwhile, “B-BODY” (5315 entities) has the f1 score of 0.745. Classification errors of the class is linked to “I-BODY” (138 FP and 288 FN). For example,

```
"胎" (tāi) ("fetus") B-BODY O
"兒" (ér) ("child") I-BODY O
"臍" (qí) ("navel") I-BODY B-BODY
"帶" (dài) ("belt") I-BODY I-BODY
```

“胎兒臍帶” (tāi ér qí dài) (“umbilical cord”) was a “BODY” sample in the test dataset. We see the Baseline system classified “胎兒” (tāi ér) (“fetus”) as “O” and “臍帶” (qí dài) (“umbilical cord”) as “BODY”. Based on statistics shown in figure 5.2, we found the most “I-BODY” is the most likely confused by “B-BODY” (0.035 percentage). Simultaneously, “B-BODY” is most often regarded as “I-BODY” (0.026 percentage). The confusion between “I-BODY” and “B-BODY” is most likely related to word ambiguity, since “臍帶” (qí dài) (“umbilical cord”) and “胎兒臍帶” (tāi ér qí dài) (“umbilical cord”) both refer to the same body part (umbilical cord). The Baseline model was overly confident when predicting “胎兒” (tāi ér) (“fetus”) as outside of the “BODY” entity.

(3) "I-SYMP" and "B-SYMP"

The “B-SYMP” class achieved 0.630 f1 measurement (1944 support). The misclassification of the class is highly correlated with “B-DISE” (65 FP), “I-DISE” (63 FP), and ”B-BODY” (92 FN). Here is an example,

```
"血" (xuè) ("blood") B-SYMP B-BODY
"液" (yè) ("fluid") I-SYMP I-BODY
"逆" (nì) ("reverse") I-SYMP B-DISE
"流" (liú) ("flow") I-SYMP I-DISE
```

“血液逆流” (xuè yè nì liú) (“blood reflux”) is a medical condition where blood flows backward through a blood vessel or heart chamber, it was annotated as a real “SYMP” entity. The baseline system treated the “血液” (xuè yè) (“blood”) and “逆流” (nì liú) (“reflux”) as “BODY” and “DISE” respectively. The errors reveal the underlying issue that many body part(s) or disease names, or even both are used for forming certain symptom names, therefore the three entities were very often confused when the model has seen part of text or the whole text being classified as the other class(es). This phenomenon is also universal since most disease words originate from the symptoms, or the other way around. For example, in Latin, “febris” (“fever”) could be used to describe the disease or the symptom.

In the meantime, “I-SYMP” (2878 samples) yielded a f1 score of 0.582. Errors of “I-SYMP” is mainly related to “I-DISE” (154 FP) and “I-BODY” (129 FN), “B-SYMP” (99 FN), and “I-DISE” (96 FN). Here is an example,

```
"精" (jīng) ("essence") B-SYMP B-BODY
"神" (shén) ("spirit") I-SYMP I-BODY
"錯" (cuò) ("wrong") I-SYMP B-SYMP
"亂" (luàn) ("disorder") I-SYMP I-SYMP
```

“精神錯亂”(jīng shén cuò luàn) (“mental confusion or psychosis”) was regarded as different entities by the Baseline system: “精神” (jīng shén) (“mental”) as “BODY” and “錯亂” (cuò luàn) (“confusion”) as “SYMP.” As demonstrated in figure 5.2, we see more “B-SYMP” (0.047 percentage) classification errors are related to “B-BODY”, while higher ratio of “I-SYMP” (0.045 percentage) was classified incorrectly as “I-BODY”.

5.1.2 DRUG-CHEM-SUPP

(1) "I-DRUG" and "B-DRUG"

The “I-DRUG” (1599 cases) class has the f1 score of 0.667, the majority of misclassification was related to “I-CHEM” (99 FP) and “B-BODY” (67 FN). For example,

```
"皮" (pí) ("skin") B-DRUG B-BODY
"質" (zhì) ("quality") I-DRUG B-CHEM
"類" (lèi) ("type") I-DRUG I-CHEM
"固" (gù) ("solid") I-DRUG I-CHEM
"醇" (chún) ("alcohol") I-DRUG I-CHEM
```

“皮質類固醇”(pí zhì lèi gù chún) (“corticosteroids”) was annotated as a type of “DRUG” (steroid hormones produced in the adrenal cortex) based on presence of “氣喘” (qì chuǎn) (“asthma”) in previous text. The Baseline model identified the text as two different entities: “皮” (pí) (“skin”) as a “BODY” entity and “質類固醇” (zhì lèi gù chún) (“steroids”) as a “CHEM” entity. The confusion of “CHEM” and “DRUG” is likely related to “質類固醇” (zhì lèi gù chún) (“steroids”) being a ”multi-entity” word since the word can be categorized as “CHEM” or “DRUG” according to the context.

Similarly, “B-DRUG” (481 support) obtained a f1 score of 0.583. “B-CHEM” (17 FP) and “B-CHEM” (35 FN) was most correlated to the classification errors. As an instance,

```
"感" (gǎn) ("sense") B-DRUG B-CHEM
"冒" (mào) ("risk") I-DRUG I-CHEM
"糖" (táng) ("sugar") I-DRUG I-CHEM
"漿" (jiāng) ("paste") I-DRUG I-CHEM
```

"感冒糖漿"(gǎn mào táng jiāng) ("coughing syrup") is a "DRUG" entity. The Baseline system, considered it a "CHEM" entity, despite the not fitting the description of the "CHEM" entities. We were convinced it is an error due to training samples. According to figure 5.2, "B-CHEM" (0.073 percentage) and "I-CHEM" (0.076 percentage) are the classes with the highest percentage of misclassification for "B-DRUG" "I-DRUG" entities.

(2) "I-CHEM" and "B-CHEM"

The "I-CHEM" class (3851 samples) has the f1 score of 0.717, with more classification errors seen among "I-BODY" (124 FP) and "I-DRUG" (122 FN). For example:

```
"假" (jiǎ) ("fake") B-CHEM B-DRUG
"麻" (má) ("numb") I-CHEM I-DRUG
"黃" (huáng) ("yellow") I-CHEM I-DRUG
"素" (sù) ("element") I-CHEM I-DRUG
"鹼" (jiǎn) ("alkali") I-CHEM O
```

"假麻黃素鹼" (jiǎ má huáng sù jiǎn) ("ephedrine alkaloid") refers to a class of "CHEM" (chemical compounds derived from ephedra plants). The text was categorized as a "DRUG" entity and "O" by the Baseline. It is possibly a classification mistake caused by "multi-entity" word (the text can be categorized as "DRUG" or "CHEM") or an annotation error, i.e., " 鹼" (jiǎn) ("alkali") is actually a part of the "CHEM" entity.

At the same time, "B-CHEM" (1718 samples) has a f1 score of 0.644, with false classification mostly related to "I-CHEM" (99 FP) and "B-BODY" (67 FN). For example,

```
"藥" (yào) ("medicine") B-CHEM O
"物" (wù) ("object") I-CHEM O
"酵" (jiào) ("ferment") I-CHEM B-CHEM
"素" (sù): ("element") I-CHEM I-CHEM
```

"藥物酵素" (yào wù jiào sù) ("medical enzyme") is a "CHEM" entity (pharmaceuticals typically used for drug formulations). The Baseline identified the text as "O" and "CHEM" for "藥物"(yào wù) ("drug") and "酵素"(jiào sù) ("enzyme") respectively. The error could be caused by lack of context information. Based on statistics from figure 5.2, we see the class with the highest relevance of misclassification of "CHEM" entities being "BODY", with 0.032 percentage of "I-CHEM" and 0.039 percentage of "B-CHEM" for "I-BODY" and "B-BODY".

(3) "I-SUPP" and "B-SUPP"

"I-SUPP" (551 samples) has the f1 measurement of 0.729. The most frequent errors are "I-CHEM" (47 FP) and "B-SUPP" (47 FN). For example,

```
"核" (hé) ("Core") B-SUPP B-CHEM
"黃" (huáng) ("yellow") I-SUPP I-CHEM
"素" (sù) ("element") I-SUPP I-CHEM
```

“核黃素”(hé huáng sù) (“riboflavin”), also known as vitamin B2, is a “SUPP” entity. The Baseline classifier considered it a ”CHEM” entity, since “核黃素” (hé huáng sù) (“riboflavin”) can be seen as a “multi-entity” instance which complicates the classification.

Distinguishing certain chemistry and supplement in Chinese could be challenging. This is also the case for other languages such as German, e.g., “Eisen” refers to the chemical element iron, as well as the iron supplements. Therefore, the boundaries between the two classes hinges on the samples that the model learned from training.

“B-SUPP” (183 instances) achieved a f1 score of 0.706. Primarily due “B-CHEM” (16 FN). After examining figure 5.2, we found “I-SUPP” (0.085 percentage) and “B-SUPP” (0.087 percentage) are regarded as the “I-CHEM” and “B-CHEM” entities.

5.1.3 Other Classes

Apart from the classes mentioned above, we also see relevance of the “TREAT” and “EXAM” highly linked to “BODY”. It is hardly surprising considering there are many instances of medical check or treatment that use the human body part for naming.

“I-TREAT” (1251 support) has a f1 score of 0.643. The errors mainly involve “I-BODY” (77 FN), meanwhile, “B-TREAT” (468 samples) has the F1 score of 0.643, with “I-TREAT” (39 FP) and“B-BODY” (60 FN). As an example,

```
"肩" (jiān) ("shoulder") B-TREAT B-BODY
"關" (guān) ("joint") I-TREAT I-BODY
"節" (jié) ("joint") I-TREAT I-BODY
"囊" (náng) ("capsule") I-TREAT I-BODY
"擴" (kuò) ("enlarge") I-TREAT O
"張" (zhāng) ("spread") I-TREAT O
"術" (shù) ("technique") I-TREAT O
```

We see the Baseline system recognized “肩關節囊” (jiān guān jié náng) (“shoulder joint capsule”) and “擴張術” (kuò zhāng shù) (“expansion surgery”) as “BODY” and “O” separately. Using body part for naming clinical treatments is also very popular for languages other than Chinese. For example, Italian uses he treatment name (“cervicale”) to form the entity of “fisioterapia cervicale” (“cervical physiotherapy”).

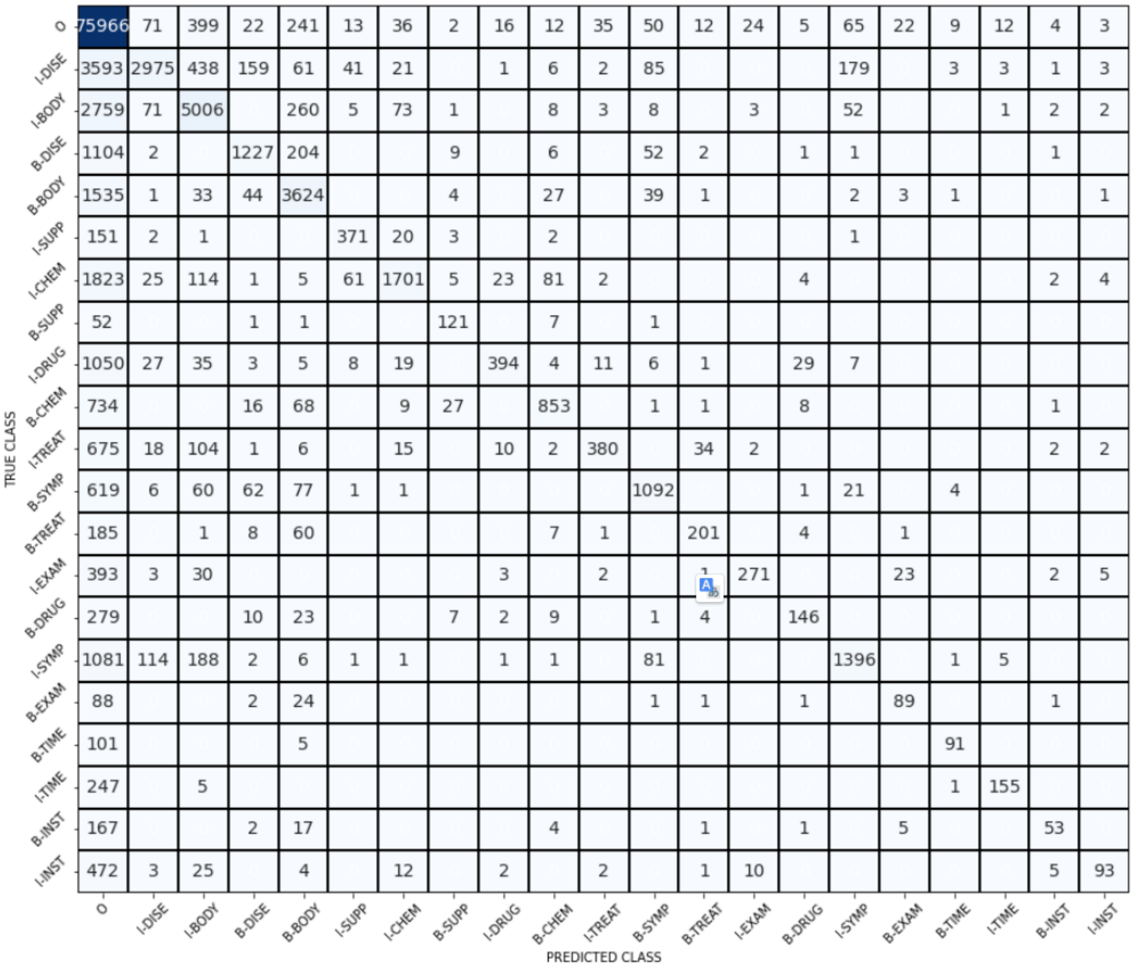
Meanwhile, “I-EXAM” (733 cases) has the overall score of 0.597 (f1), with “I-BODY” (43 FN). The classification errors of “B-EXAM” (207 instances, 0.506 F1-score) mainly concerns “I-EXAM” (30 FP) and “B-BODY” (34 FN). Here is an example,

```
"心" (xīn) ("heart") B-EXAM B-BODY
"臟" (zàng) ("gut") I-EXAM I-BODY
"負" (fù) ("negative") I-EXAM O
"荷" (hè) ("lotus") I-EXAM O
"測" (cè) ("test") I-EXAM O
"試" (shì) ("experiment") I-EXAM O
```

“心臟負荷測試” (xīn zàng fù hè cè shì) (“cardiac stress test”) is a medical procedure used to evaluate the heart’s ability to respond to stress or exercise. The Baseline categorized “心臟” (xīn zàng) (“heart”) as “BODY” and “O” for the rest of the text. This error primarily concerns the lack of context since “心臟” (xīn zàng) (“heart”) is an instance of the “BODY” class. Without knowing the preceding text, the classifier is easy to misjudge the characters as a standalone “BODY” entity.

The prediction of the Baseline system on “B-TIME” (197), “I-TIME” (408), “B-INST” (250), “I-INST” (629) demonstrated fine precision (0.908, 0.90, 0.821, 0.883), but much worse recall scores (0.350, 0.267, 0.092, 0.084). The low recall scores hurt the performance severely, leading to the F1 scores of 0.505, 0.412, 0.165, 0.154, respectively.

5.2 Jieba Full (word-based, Epoch = 5) on test data

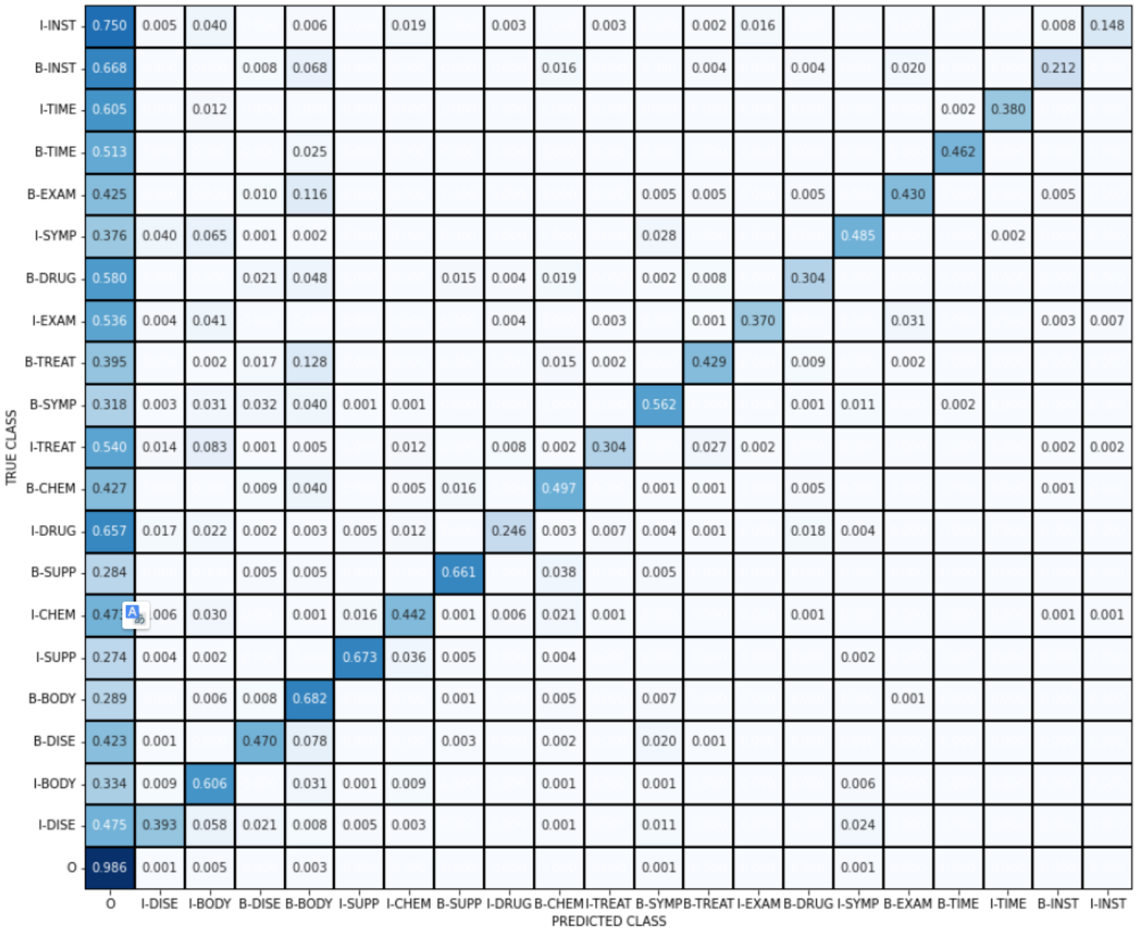


	O	I-DISE	I-BODY	B-DISE	B-BODY	I-SUPP	I-CHEM	B-SUPP	I-DRUG	B-CHEM	I-TREAT	B-SYMP	B-TREAT	I-EXAM	B-DRUG	I-SYMP	B-EXAM	B-TIME	I-TIME	B-INST	I-INST
O	75966	71	399	22	241	13	36	2	16	12	35	50	12	24	5	65	22	9	12	4	3
I-DISE	3593	2975	438	159	61	41	21	0	1	6	2	85	0	0	0	179	0	3	3	1	3
I-BODY	2759	71	5006	0	260	5	73	1	0	8	3	8	0	3	0	52	0	0	1	2	2
B-DISE	1104	2	0	1227	204	0	0	9	0	6	0	52	2	0	1	1	0	0	0	1	0
B-BODY	1535	1	33	44	3624	0	0	4	0	27	0	39	1	0	0	2	3	1	0	0	1
I-SUPP	151	2	1	0	0	371	20	3	0	2	0	0	0	0	0	1	0	0	0	0	0
I-CHEM	1823	25	114	1	5	61	1701	5	23	81	2	0	0	0	4	0	0	0	0	2	4
B-SUPP	52	0	0	1	1	0	0	121	0	7	0	1	0	0	0	0	0	0	0	0	0
I-DRUG	1050	27	35	3	5	8	19	0	394	4	11	6	1	0	29	7	0	0	0	0	0
B-CHEM	734	0	0	16	68	0	9	27	0	853	0	1	1	0	8	0	0	0	0	1	0
I-TREAT	675	18	104	1	6	0	15	0	10	2	380	0	34	2	0	0	0	0	0	2	2
B-SYMP	619	6	60	62	77	1	1	0	0	0	0	1092	0	0	1	21	0	4	0	0	0
B-TREAT	185	0	1	8	60	0	0	0	0	7	1	0	201	0	4	0	1	0	0	0	0
I-EXAM	393	3	30	0	0	0	0	0	3	0	2	0	1	271	0	0	23	0	0	2	5
B-DRUG	279	0	0	10	23	0	0	7	2	9	0	1	4	0	146	0	0	0	0	0	0
I-SYMP	1081	114	188	2	6	1	1	0	1	1	0	81	0	0	0	1396	0	1	5	0	0
B-EXAM	88	0	0	2	24	0	0	0	0	0	0	1	1	0	1	0	89	0	0	1	0
B-TIME	101	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	91	0	0	0
I-TIME	247	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	155	0	0
B-INST	167	0	0	2	17	0	0	0	0	4	0	0	1	0	1	0	5	0	0	53	0
I-INST	472	3	25	0	4	0	12	0	2	0	2	0	1	10	0	0	0	0	0	5	93

Figure 5.3: Heatmap of the Jieba Full system on test data (word-based, Epoch=5)

Figure 5.3 and figure 5.4 provide details about Jieba Full system’s prediction errors on the test dataset. We follow the introduction style from previous section to highlight the most frequent errors in relation to corresponding predictive class(es), taking both heatmaps into account.

The Jieba Full system exhibited high capability in predicting the most prevalent class “O” (77019 instances), with the F1 score of 0.893. However, we put the “O” class aside for more comprehensive understanding of the model’s prediction errors.



	O	I-DISE	I-BODY	B-DISE	B-BODY	I-SUPP	I-CHEM	B-SUPP	I-DRUG	B-CHEM	I-TREAT	B-SYMP	B-TREAT	I-EXAM	B-DRUG	I-SYMP	B-EXAM	B-TIME	I-TIME	B-INST	I-INST
I-INST	0.750	0.005	0.040		0.006		0.019		0.003		0.003		0.002	0.016						0.008	0.148
B-INST	0.668			0.008	0.068					0.016			0.004		0.004		0.020			0.212	
I-TIME	0.605		0.012															0.002	0.380		
B-TIME	0.513				0.025													0.462			
B-EXAM	0.425			0.010	0.116							0.005	0.005		0.005		0.430			0.005	
I-SYMP	0.376	0.040	0.065	0.001	0.002							0.028				0.485			0.002		
B-DRUG	0.580			0.021	0.048			0.015	0.004	0.019		0.002	0.008		0.304						
I-EXAM	0.536	0.004	0.041						0.004		0.003		0.001	0.370			0.031			0.003	0.007
B-TREAT	0.395		0.002	0.017	0.128					0.015	0.002		0.429		0.009		0.002				
B-SYMP	0.318	0.003	0.031	0.032	0.040	0.001	0.001					0.562			0.001	0.011		0.002			
I-TREAT	0.540	0.014	0.083	0.001	0.005		0.012		0.008	0.002	0.304		0.027	0.002						0.002	0.002
B-CHEM	0.427			0.009	0.040		0.005	0.016		0.497		0.001	0.001		0.005					0.001	
I-DRUG	0.657	0.017	0.022	0.002	0.003	0.005	0.012		0.246	0.003	0.007	0.004	0.001		0.018	0.004					
B-SUPP	0.284			0.005	0.005			0.661		0.038		0.005									
I-CHEM	0.47[illegible]	0.006	0.030		0.001	0.016	0.442	0.001	0.006	0.021	0.001				0.001					0.001	0.001
I-SUPP	0.274	0.004	0.002			0.673	0.036	0.005		0.004						0.002					
B-BODY	0.289		0.006	0.008	0.682			0.001		0.005		0.007					0.001				
B-DISE	0.423	0.001		0.470	0.078			0.003		0.002		0.020	0.001								
I-BODY	0.334	0.009	0.606		0.031	0.001	0.009			0.001		0.001				0.006					
I-DISE	0.475	0.393	0.058	0.021	0.008	0.005	0.003			0.001		0.011				0.024					
O	0.986	0.001	0.005		0.003							0.001				0.001					

Figure 5.4: Normalized Heatmap of the Jieba Full system on test data (word-based, Epoch=5)

5.2.1 DISE-BODY-SYMP

```
(1) "I-DISE" and "B-DISE"
```

“B-DISE” (2609 samples) achieved a F1 score of 0.589, with misclassification primarily concerning “I-DISE” (159 FP), “B-SYMP” (114 FP) and “B-BODY” (204 FN). For an instance,

```
"心" (xīn) ("heart") B-DISE B-SYMP
"律" (lǜ) ("beat") I-DISE I-SYMP
"不" (bù) ("no") I-DISE I-SYMP
"整" (zhěng) ("whole") I-DISE I-SYMP
```

“心律不整”(xīn lǜ bù zhěng) (“Arrhythmia”) is a multi-entity that potentially related to both the “DISE” and “SYMP” classes. When we refer to the annotation guidelines, we found that symptoms are often the aftermaths of diseases, therefore, the confusion between “DISE” and “SYMP” entities is presumably sourced from the training samples.

Another potential factor is the model failed to capture more context information, which led to incorrect judgement.

"I-DISE" (7571 instances) has the overall score of 0.546 (f1), mainly due to "I-SYMP" (114 FP), "I-BODY" (438 FN), "I-SYMP" (179 FN), and "B-DISE" (159 FN). For example,

```
"血" (xuè) ("blood") B-DISE B-CHEM
"色" (sè) ("color") I-DISE I-CHEM
"素" (sù) ("element") I-DISE I-CHEM
"沉" (chén) ("sink") I-DISE B-DISE
"著" (zhe) ("continuously") I-DISE I-DISE
"病" (bìng) ("sickness") I-DISE I-DISE
```

The Jieba Full classifier divided "血色素沉著病" (xuè sè sù chén zhe bìng) ("Hemoglobin downfall sickness") into "血色素" (xuè sè sù) ("Hemoglobin") and "沉著病" (chén zhe bìng) ("downfall syndrome"), with corresponding "CHEM" and "DISE" labels. The error is potentially linked to the word segmentation strategy as well as the multi-entity "血色素" (xuè sè sù) ("Hemoglobin"), which is not only a human body part, but only used to form specific type of blood disease.

Based on figure 5.4, classes with the highest relevance to classification errors of "I-DISE" and "B-DISE" are "I-BODY" (0.058 percentage) "B-BODY" (0.078 percentage).

(2) "I-BODY" and "B-BODY"

The "B-BODY" (5315 cases) class has the second place in overall score (f1 0.724) among all entity types. The classification mistakes mostly centered on "I-BODY" (260 FP), "B-DISE" (204 FP), and "I-SYMP" (39 FN). For example,

```
"脂" (zhī) ("oil") B-BODY B-DISE
"肪" (fáng) ("fat") I-BODY I-DISE
"基" (jī) ("foundation") I-BODY B-BODY
"因" (yīn) ("because") I-BODY I-BODY
```

"脂肪基因" (zhī fáng jī yīn) ("fat genes") was split into two parts by the Jieba Full system: "脂肪" (zhī fáng) ("fat") and "基因" (jī yīn) ("gene"). They were labled as "DISE" and "BODY" entities. The prediction error is most likely due to word segmentation standard or word ambiguity (text components can be interpreted differently and still make sense).

Meanwhile, "I-BODY" (8254 cases) secures the fourth position in F1 score (0.681) among the predictive classes, with "B-BODY" (260 FN), "I-DISE" (438 FP), "I-SYMP" (188 FP), and "I-CHEM" (114 FP). For example,

```
"內" (nèi) ("inside") B-BODY B-BODY
"分" (fēn) ("divide") I-BODY I-BODY
"泌" (mì) ("discharge") I-BODY I-BODY
"細" (xì) ("thin") I-BODY B-BODY
"胞" (bāo) ("cell") I-BODY I-BODY
```

The text "內分泌細胞" (nèi fēn mì xì bāo) ("endocrine cells") was treated as "內分泌" (nèi fēn mì) ("endocrine") and "細胞" (xì bāo) ("cell") separately by the Jieba Full system. This is also likely an error caused by the word segmentation strategy.

According to figure 5.4, classification errors of "B-BODY" and "I-BODY" are mostly related to "B-DISE" (0.008 percentage) and "B-BODY" (0.031 percentage).

(3) "I-SYMP" and "B-SYMP"

The “B-SYMP” class (1944 cases) has the f1 score of 0.650. The classifier mainly made mistakes on “I-DISE” (85 FP), “I-SYMP” (81 FP), “B-BODY” (77 FN), and “B-DISE” (62 FN). Here is an example.

```
"免" (miǎn) ("free") B-SYMP B-BODY
"疫" (yì) ("epidemic") I-SYMP I-BODY
"反" (fǎn) ("opposite") I-SYMP I-BODY
"應" (yìng) ("respond") I-SYMP I-BODY
```

“免疫反應” (miǎn yì fǎn yìng) (“immune response”) was predicted as a “BODY” entity instead of “SYMP”. We can reason that the classification error is caused by the WEs used to generalize things across similar vocabulary failed to generalize well on “免疫系統” (miǎn yì xì tǒng) (“immune system”) and “免疫細胞” (miǎn yì xì bāo) (“immune cells”).

Moreover, “I-SYMP” (2878 samples) secured the overall score of 0.607 (f1). We found “I-DISE” (179 FP), “I-BODY” (188 FN), “I-DISE” (114 FN) are the most frequently errors concerning the predictive class.

Based on figure 5.4, the most misclassified classes of “B-SYMP” and “I-SYMP” are “B-BODY” (0.008 percentage) and “I-BODY” (0.06 percentage).

5.2.2 DRUG-CHEM-SUPP

(1) "I-DRUG" and "B-DRUG"

Moving on, “B-DRUG” (481 instances) and “I-DRUG” (1599 cases) obtained considerably high precision (0.730, 0.872), but both categories suffer from low recall (0.304 and 0.246), which led to the overall score of 0.429 and 0.384 (f1 scores). Classification errors of “B-DRUG” concerns “I-BODY” (35 FN) “B-DRUG” (29 FN), while that of “I-DRUG” mainly involves “I-CHEM” (16 FP), “I-DRUG” (23 FP), and “B-DRUG” (29 FN). As an instance,

```
"口" (kǒu) ("oral") O B-DRUG
"服" (fú) ("serve") O I-DRUG
"抗" (kàng) ("resist") B-DRUG I-DRUG
"生" (shēng) ("raw") I-DRUG I-DRUG
"素" (sù) ("element") I-DRUG I-DRUG
```

“口服抗生素” (kǒu fú kàng shēng sù) (“drinkable antibiotic”) was segmented as “口服” (kǒu fú) (“apply orally”) and “抗生素” (kàng shēng sù) (“antibiotic”) by the annotators, with the judgement that the two parts are “O” and “DRUG” entities. However, when we refer to the guidelines, we discovered the text is much better fit for the description of “DRUG”. Therefore, we are convinced this error is a case of annotation mistake, for example, the annotators agreed on the majority vote as the gold label.

According to figure 5.4, the most misclassified classes of “B-DRUG” and “I-DRUG” are “B-BODY” (0.048 percentage) and “B-DRUG” (0.018 percentage).

(2) "I-CHEM" and "B-CHEM"

For the “B-CHEM” class (1718 samples) class, Jieba Full yielded a moderate performance (0.621 f1 score). Primarily due to ‘I-CHEM” (81 FP), “B-BODY” (27 FP), “I-CHEM” (734 FN), and “B-DISE” (68 FN). Here is an example,

```
"幽" (yōu) ("dark") B-CHEM B-BODY
"門" (mén) ("door") I-CHEM I-BODY
"螺" (luó) ("screw") I-CHEM O
"旋" (xuán) ("rotate") I-CHEM O
"桿" (gǎn) ("stick") I-CHEM O
"菌" (jūn) ("fungus")I-CHEM O
```

“幽門螺旋桿菌” (yōu mén luó xuán gǎn jūn) (“Helicobacter pylori”) is a “CHEM” entity mistreated by the Jieba Full model as “BODY” and “O”, in correspondence with “幽門” (yōu mén) (“the first part of the small intestine”) and “螺旋桿菌” (luó xuán gǎn jūn) (“helicobacter pylori”). The error is likely caused by the word segmentation strategy.

Meanwhile, “I-CHEM” (3851 samples) has the f1 score of 0.591. Looking more closely, we discovered “I-BODY” (73 FP) and “I-BODY” (114 FN) are the most frequent errors of the predictive class. For instance,

```
"水" (shuǐ) ("water") B-CHEM O
"解" (jiě) ("relieve") I-CHEM O
"酵" (jiào) ("ferment") I-CHEM B-CHEM
"素" (sù) ("element") I-CHEM I-CHEM
```

“水解酵素” (shuǐjiě jiàosù) (“hydrolytic enzyme”) was split into “水解” (shuǐ jiě) (“hydrolytic”) and “酵素” (jiàosù) (“enzyme”), with the predicted entity types “O” and “CHEM”. It is also seen as an error of word separation standard.

Based on figure 5.4, “B-CHEM” and “I-CHEM” misclassification is mainly related to “B-SUPP” (0.016 percentage) and “I-BODY” (0.030 percentage).

(3) "I-SUPP" and "B-SUPP"

“I-SUPP” (551 cases) achieves an F1 score of 0.705, with notable errors revolving “I-CHEM” (61 FP), “I-DISE” (41 FP) and “I-CHEM” (20 FN). We use the following text as an example:

```
"生" (shēng) ("raw") B-SUPP B-CHEM
"物" (wù) ("matter") I-SUPP I-CHEM
"素" (sù) ("element") I-SUPP I-CHEM
```

“生物素” (shēng wù sù) (“biological elements”) is a “SUPP” entity treated as “CHEM” by the Jieba Full classifier. We relate the error to the text is a “multi-class” entity that can be categorized as “SUPP” or “CHEM” based on the definition of the entities.

The “B-SUPP” (183 instances) class achieved F1 score of 0.669, with classification error mostly concerning the “B-CHEM” (27 FP) and “B-CHEM” (7 FN) classes.

As shown in figure 5.4, incorrect predictions of “I-SUPP” and “B-SUPP” are mostly distributed in “I-CHEM” (0.036 percentage) and “B-CHEM” (0.038 percentage).

5.2.3 Other Classes

Interestingly, the overall scores of “B-TIME” (197 examples) and “I-TIME” (408 instances) turned out to be above average (0.593 and 0.531 f1 scores). We were convinced this is due to limited training instances.

“B-TREAT” (468 samples) displayed a below average f1 score (0.552) compared to other predictive classes. mostly dragged by “B-TREAT” (34 FP) and “B-BODY” (60 FN). “I-TREAT” (1251 examples) has a lower overall score (0.450) compared with “B-TREAT” due to much lower (0.429 compared to 0.304), despite a higher precision (0.868 compared to 0.773). According to figure 5.4, the classification error mostly concentrated on “B-BODY” (0.128 percentage) and “I-BODY” (0.083 percentage) for “B-TREAT” and “B-TREAT”. For example,

```
"乳" (rǔ) ("diary") B-TREAT B-BODY
"房" (fáng) ("house") I-TREAT I-BODY
"切" (qiè) ("cut") I-TREAT O
"除" (chú) ("remove") I-TREAT O
"術" (shù) ("skill") I-TREAT O
```

shows the wrong word segmentation decision of cutting “乳房切除術” (rǔ fáng qiè chú shù) (“mastectomy”) into “乳房” (rǔ fáng) (“breasts”) and “切除術” (qiè chú shù) (“resection”), which led to the predictions of “BODY” and “O”.

“I-EXAM” (733 instances) and “B-EXAM” (207 instances) have the close f1 scores of 0.520 and 0.509. Lastly, the “B-INST” (250 entities) and “I-INST” (629 entities) have the low F1 scores of 0.327 and 0.251.

5.3 Comparing the two systems

We compare the Baseline system (character-based 3 epochs) and the Jieba Full system (word-based 5 epochs) based on the f1 scores presented in table 4.3 and table 4.4 across different predictive classes.

Despite the gap between the overall performance of Baseline (0.612 f1-score) and the Jieba Full (0.563 f1-score), we still see potential in using word-level input for training the BiLSTM-CRF model. Since Jieba Full demonstrated strong effort in capturing classes with much smaller sample sizes where the f1 scores witnessed significant growth, for example, the f1 measurement of “B-SYMP” went from 0.630 (Baseline) to 0.650 (Jieba Full), those of “I-SYMP” went from 0.582 (Baseline) to 0.607 (Jieba Full). Moreover, we see the overall scores of “B-TIME”, “I-TIME”, “B-INST”, “B-EXAM”, and “I-INST” all witnessed different levels of boost. Mostly owing to the considerably improvement of recall scores. Except for “B-EXAM”, where the impact of the precision rise (from 0.595 to 0.622) exceeded the recall drop (from 0.440 to 0.430.) on the f1 score.

The Baseline system outperformed the Jieba Full model in predicting most entities. Lower recall scores is the primary reason of the situation. For example, Baseline yielded the precision and recall of 0.826 and 0.721, while those of the Jieba Full system were 0.777 and 0.606.

Both systems face the same struggle of maintaining a balance between precision and recall metrics. Both systems exhibited fine generalization on certain entity types but struggles with maintaining the performance on a similar level across other classes.

5.3.1 Potential Factors of the Errors

In terms of the causes of prediction errors, we group them into the following categories:

(1) Limited training samples

The low-resource class(es) create(s) difficulty for machine learning algorithms to find pattern(s) from insufficient data.

(2) Confusion between certain classes

Classes such as “DISE-BODY-SYMP”, “DRUG-CHEM-SUPP”, “BODY-TREAT-EXAM” can be confusing due to the existence of “multi-entity” or “multi-class” instances. For example, “瘤” (liú) (“tumor”) can be seen as a part of the human body (“BODY” entity), but when it occurs in “有腫瘤” (yǒu zhǒng liú) (“having the tumor”), the text is very obvious a mention of the “SYMP” or “DISE” entity. Therefore, the character “瘤” (liú) (“tumor”) is no longer labeled as “BODY”. This linguistic phenomenon (disease entity borrows word from body part) is not limited in the Chinese language, but also other languages such as English. For example, “heart” is a muscular organ (“BODY”), but it can also compose the expression of a ”heart disease” (e.g., heart failure, a “DISE” entity).

The belonging of these entities are transformable between classes since they borrow words from each other, which poses challenges for classification task like CHNER where only one label is allowed for each text instance.

(3) Human annotations

The quality of annotation is another issue that may hinder the validation accountability. After examining the gold labels carefully, we found questionable annotations which conflict with the definition of entity type. For example,

```
"由" (yóu) ("from") O O
"脂" (zhī) ("oil") B-BODY B-CHEM
"肪" (fáng) ("fat") I-BODY I-CHEM
"細" (xì) ("slim") I-BODY B-BODY
"胞" (bāo) ("flesh") I-BODY I-BODY
"組" (zǔ) ("organize") O O
"成" (chéng) ("into") O O
"，"        (comma) O O
"用" (yòng) ("use") O O
"來" (lái) ("come") O O
"儲" (chǔ) ("save") O O
"存" (cún) ("save") O O
"脂" (zhī) ("oil") B-CHEM B-CHEM
"肪" (fáng) ("fat") I-CHEM I-CHEM
```

“脂肪” (zhī fáng) (“fat”) was annotated as “BODY” and “CHEM” in the same sentence, after checking the annotation guidelines, we found that it is likely a annotation error since based on the context (“Adipose cells are composed of fat, and used to store, fat”), we can safely assume the second mention of “fat” still refer to the fat in human body.

Therefore, the decision of how to annotate the data could take this into consideration, e.g., allowing a couple possible sets of annotation for the same entities. Or using majority vote to settle disagreement between annotators in exchange for a more satisfying IAA score.

(4) Word Boundary Ambiguity

The word boundary ambiguity concerns words with varying semantic interpretations based on the context, for example, “角质細胞老化” (jué zhì xì bāo lǎo huà) (“keratin cellular aging”) can be viewed as a single entity, as well as two parts, i.e., “角质” (jué zhì) (“keratin”) and “細胞老化” (xì bāo lǎo huà) (“cellular aging”). Therefore, the text is easily recognized as different classes when treated independently. We saw many examples of “DISE” treated as “BODY + DISE” or “BODY + SYMP” due the blurry word boundaries. Therefore, word boundary ambiguity can lead to classification error, especially without sufficient information of the surrounding of text.

Chapter 6

Discussion

This section examine the experiment discoveries outlined in 4 by taking a closer look at the obstacles and constraints of our research inquiries and pinpoint solutions in future work that could potentially elevate the BiLSTM-CRF model's performance on the CHNER task.

6.1 Experiment Discoveries

Our experiment is determined to find out the question of:

- Whether word-based systems are better than character-based systems (using the same BiLSTM-CRF model) on predicting Chinese healthcare entities?

We found that:

- Word-based system(s) are not necessarily more competitive than the character-based system(s) in predicting CHNER labels.

We utilized domain knowledge from existing corpus (*CHNER*), i.e., character-based data ("CHNER Char") and external resources to create three additional versions of the original data, i.e., word-based data ("Jieba Base", "Jieba Upgrade", "Jieba Full"). The "CHNER Char" datasets parsed the characters from the *CHNER* datasets. The three word-based datasets were created by *Jieba* supplied by different dictionaries ("default", "ROLING", "FUSION"). The training-dev data saved with the gold labels in the same format as the"CHNER Char" datasets. While the test set was strictly confined to the text (sub-tokens or characters.)

We trained WEs that match the word segmentation strategies of the task data (train-dev-test set), the training corpus was formed by training-dev sets of the CHNER datasets. We used the same BiLSTM-CRF base model for training the classifiers. After configuring the epoch numbers, we find the best word-based and character-based models according to the performance on the development data. The performance of the selected systems on the test data were measured by the standard NER evaluation metrics (precision, recall, f1).

The "Baseline" (character-based 3 epochs) system yielded a 0.612 f1 score. The "Jieba Full" (word-based 5 epochs) system achieved the reasonable F1 score of 0.563. With lower precision (0.798) and recall (0.464) compared to those of the Baseline (P:0.831, R:0.621).

A point worth mentioning is the “Jieba Full” has demonstrated great ability in predicting less supported entities such as “B-EXAM” (207) “B-TIME” (197), “I-TIME” (408), “B-INST” (250), “I-INST (629). We believe the Jieba Full model’s full potential was yet to be discovered. Primarily due to the training stopped at 7 epochs when the overall score slowed down. It is unknown whether it is the best Epoch value if further configuration was to be implemented. Besides, switching the labels between word level and character level has the underlying risk of information loss, for example,

```
character-based text and gold labels:
血 (xuè) ("blood") B-DISE
色 (sè) ("color") I-DISE
素 (sù) ("element") I-DISE
沉 (chén) ("sink") B-DISE
著 (zhe) ("continuously") I-DISE
病 (bìng) ("sickness") I-DISE

word-based text and gold labels:
血色素沉著病 B-DISE
(xuè sè sù chén zhe bìng)
("Hemoglobin downfall sickness")

restored text and gold lables on character level:
血 (xuè) ("blood") B-DISE
色 (sè) ("color") I-DISE
素 (sù) ("element") I-DISE
沉 (chén) ("sink") I-DISE
著 (zhe) ("continuously") I-DISE
病 (bìng) ("sickness") I-DISE
```

When the text was treated as a whole entity by the segmentation scheme, the merging of gold labels changes the original annotation of the characters, despite the restored labels align with the original text in length (number of characters). When the contradictory gold labels were used as example(s) for training the CHNER classifier. It is very likely that the trained classifier will treat the same text as a single entity instead of two. For such reason, merging the gold labels has the underlying issue of changing the training instances of the word-based systems.

Through error analysis, we realize that CHNER is a complex NLP problem yet to be fully addressed. The prediction errors mainly concern the following factors: (1) Limited training samples; (2) Confusion between certain classes; (3) Word boundary Ambiguity; (4) Annotation Errors.

6.2 Limitations and Future work

Based on the experiment discoveries, we reflect on the limitations and provide possible solutions in future work:

(1) Quality of the self-defined dictionaries

Despite the segmentation results show the size of Jieba dictionaries are getting bigger (and hopefully better) as we expanded the domain lexicon on a large scale. However, the method of creating these dictionaries was fairly naive and there are likely many problems/noise with the data. For example, there are vocabularies that are not relevant to the task. Or the vocabulary is more inclined to be multiple entities instead of one single entity sample. Therefore, crafting better (cleaner) dictionaries could very likely help the method further. It is the first and most important limitation of our experiment. We need to improve the quality of the dictionary used for word segmentation, for example, through humanly inspecting the lexicon, or using more reliable resources.

`(2) Word embeddings (WEs)`

Another limitation of our experiment is that we trained the WEs only on the training-development data. This is often not enough for generalizing a more robust and generic embedding space for vector representation. Both static (e.g. Word2Vec) and contextual (e.g. BERT) embeddings using lots of data for pre-training the WEs (We did not use contextual embeddings since it leaves out the possibility of training WEs.) Some of the classification mistakes are very likely related to how the WEs were learned (although this hypothesis was not tested.) But based on the very limited training data for training the WEs, we can not expect the language model to generalize too well for unseen data. Therefore, in future attempts, we should add a lot more data (including training-development sets of the task data) to create a better embedding space for the model.

Overall, our proposed systems showed potential in handling the CHNER task, despite there is still a gap between our best system and the wining team of the shared task. We believe after deploying the above mentioned methods, our system can be improved in future work.

Chapter 7

Conclusion

This thesis inspected the effectiveness of using domain-specific vocabulary to implement different word segmentation strategies (word-based instead of character-based) on the medical data (CNHER datasets) to train the same language model (BiLSTM-CRF) in the attempt to bring out more competitive performance on the CHNER task.

The prepared training datasets (train-dev) are also used for creating matching WEs by applying different word segmentation strategies. The trained classifiers first evaluated on the development data for different epoch numbers (3, 5, 7) to find the most fit ones. Afterwards, one word-based model is picked from the rest based on the results (macro f1-score) on development data. Finally, the character-based model ("Baseline") is compared with the word-based model ("Jieba Full") against the test data to find out what segmentation scheme presents better performance (macro F1 score) on the CHNER task. The Baseline model outperformed the Jieba Full model, with the f1 measurement of 0.612. Despite the discoveries of the experiment, we still question whether character-based input is better than word-based input based on the classification errors we came across in chapter 5. Besides, there is evidence that proves the strengths of using word-based input data in predicting entities with less supporting instances. We believe there is still room for improving the word-based model in future work.

Appendix A

Appendix

Dataset	Description
Chinese medical dialogue dataset (Huatuo-26M)	the largest traditional Chinese medicine (TCM) question-answering dataset to date
cMedQA2 (108K)	A medical Q&A dataset in Chinese, with over 100,000 entries.
39Health-KG (210K triples)	Includes approximately 37,000 entities, and 210,000 entity relationships
Medical-Dialogue-System	The MedDialog dataset (Chinese) contains conversations between doctors and patients. The dataset includes 1.1 million dialogues and 4 million sentences
Yidu-S4K	Named entity recognition, entity and attribute extraction.
Yidu-N7K	Chinese medical Q&A dataset
Chinese medical Q&A dataset	Chinese medical doctor-patient dialogue data
CPubMed-KG (4.4M triples)	hinese Medical Association high-quality full-text journal data
Chinese medical knowledge graph CMeKG	CMeKG (Chinese Medical Knowledge Graph)
CHIP Annual Evaluation	CHIP Annual Evaluation (official evaluation)
Ruijin Hospital Diabetes Dataset	Ruijin Hospital Diabetes Dataset (Diabetes)

Table A.1: Other Chinese medical Corpus

CBLUE Datasets	Entities	Mean	Std	Task
CHIPCDN	5,432	8.0	2.4	CDN
CMeEE	40,545	6.0	2.5	NER
CMeIE	3,212	5.8	2.3	CMIE
IMCS	431	3.6	1.5	unknown

Table A.2: Entities across CBLUE Datasets (1 decimal)

Dataset(s)	Jieba Base	Jieba Upgrade	Jieba Full
Train	887,490	866,072	865,733
Dev	77,183	74,583	74,543
Test	71,915	68,904	68,613
Mean Length	1.6	1.6	1.6
Standard Deviation	0.6	0.8	0.8
Maximum Length	7	18	18

Table A.3: Segmentation Results (The upper part of the table shows the number of words or characters in train-dev-test set of each dataset, the lower part of the table shows the Mean length, standard deviation, maximum length of train-dev-test data together of each dataset)

Bibliography

J. Cheng, J. Liu, X. Xu, D. Xia, L. Liu, and V. S. Sheng. A review of chinese named entity recognition. *KSII Transactions on Internet & Information Systems*, 15(6), 2021.

S.-T. Chiou, S.-W. Huang, Y.-C. Lo, Y.-H. Wu, and J.-L. Wu. Scu-nlp at rocling 2022 shared task: Experiment and error analysis of biomedical entity detection model. In *Proceedings of the 34th Conference on Computational Linguistics and Speech Processing (ROCLING 2022)*, pages 350–355, 2022.

Y. Cui, W. Che, T. Liu, B. Qin, and Z. Yang. Pre-training with whole word masking for chinese bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3504–3514, 2021.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

C. Dong, J. Zhang, C. Zong, M. Hattori, and H. Di. Character-based lstm-crf with radical-level features for chinese named entity recognition. In *Natural Language Understanding and Intelligent Applications: 5th CCF Conference on Natural Language Processing and Chinese Computing, NLPCC 2016, and 24th International Conference on Computer Processing of Oriental Languages, ICCPOL 2016, Kunming, China, December 2–6, 2016, Proceedings 24*, pages 239–250. Springer, 2016.

Z.-Q. Feng, P.-K. Chen, and J.-C. Wang. Ncu1415 at rocling 2022 shared task: A light-weight transformer-based approach for biomedical name entity recognition. In *Proceedings of the 34th Conference on Computational Linguistics and Speech Processing (ROCLING 2022)*, pages 316–320, 2022.

G. Fu, C. Kit, and J. J. Webster. Chinese word segmentation as morpheme-based lexical chunking. *Information Sciences*, 178(9):2282–2296, 2008.

L. Gong, Z. Zhang, S. Chen, et al. Clinical named entity recognition from chinese electronic medical records based on deep learning pretraining. *Journal of healthcare engineering*, 2020, 2020.

T. Guan, H. Zan, X. Zhou, H. Xu, and K. Zhang. Cmeie: construction and evaluation of chinese medical information extraction dataset. In *Natural Language Processing and Chinese Computing: 9th CCF International Conference, NLPCC 2020, Zhengzhou, China, October 14–18, 2020, Proceedings, Part I 9*, pages 270–282. Springer, 2020.

H. He and J. D. Choi. The stem cell hypothesis: Dilemma behind multi-task learning with transformer encoders. *arXiv preprint arXiv:2109.06939*, 2021.

Z. Hongying, L. Wenxin, Z. Kunli, Y. Yajuan, C. Baobao, and S. Zhifang. Building a pediatric medical corpus: Word segmentation and named entity annotation. In *Chinese Lexical Semantics: 21st Workshop, CLSW 2020, Hong Kong, China, May 28–30, 2020, Revised Selected Papers 21*, pages 652–664. Springer, 2021.

L.-H. Lee and Y. Lu. Multiple embeddings enhanced multi-graph neural networks for chinese healthcare named entity recognition. *IEEE Journal of Biomedical and Health Informatics*, 25(7):2801–2810, 2021.

L.-H. Lee, C.-Y. Chen, L.-C. Yu, and Y.-H. Tseng. Overview of the ROCLING 2022 shared task for Chinese healthcare named entity recognition. In Y.-C. Chang and Y.-C. Huang, editors, *Proceedings of the 34th Conference on Computational Linguistics and Speech Processing (ROCLING 2022)*, pages 363–368, Taipei, Taiwan, Nov. 2022a. The Association for Computational Linguistics and Chinese Language Processing (ACLCLP). URL https://aclanthology.org/2022.rocling-1.46.

L.-H. Lee, C.-Y. Chen, L.-C. Yu, and Y.-H. Tseng. Overview of the rocling 2022 shared task for chinese healthcare named entity recognition. In *Proceedings of the 34th Conference on Computational Linguistics and Speech Processing (ROCLING 2022)*, pages 363–368, 2022b.

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

J. Li, A. Sun, J. Han, and C. Li. A survey on deep learning for named entity recognition. *IEEE transactions on knowledge and data engineering*, 34(1):50–70, 2020.

B.-S. Lin, J.-H. Chen, and T.-H. Chang. Nerve at rocling 2022 shared task: a comparison of three named entity recognition frameworks based on language model and lexicon approach. In *Proceedings of the 34th Conference on Computational Linguistics and Speech Processing (ROCLING 2022)*, pages 343–349, 2022.

T. Lin, G. Chonghui, and C. Jingfeng. Review of chinese word segmentation studies. *Data Analysis and Knowledge Discovery*, 4(2/3):1–17, 2020.

G. Liu and J. Guo. Bidirectional lstm with attention mechanism and convolutional layer for text classification. *Neurocomputing*, 337:325–338, 2019.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

R. Luo, J. Xu, Y. Zhang, Z. Zhang, X. Ren, and X. Sun. Pkuseg: A toolkit for multi-domain chinese word segmentation. *arXiv preprint arXiv:1906.11455*, 2019.

X. Luo, J. Wang, and X. Zhang. Ynu-hpcc at rocling 2022 shared task: A transformer-based model with focal loss and regularization dropout for chinese healthcare named entity recognition. In *Proceedings of the 34th Conference on Computational Linguistics and Speech Processing (ROCLING 2022)*, pages 335–342, 2022.

C. Ma, Z. Xu, M. Feng, J. Yin, L. Ruan, and H. Su. Context enhanced and data augmented w 2 ner system for named entity recognition. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 145–155. Springer, 2022a.

H.-Y. Ma, W.-J. Li, and C.-L. Liu. Migbaseline at rocling 2022 shared task: Report on named entity recognition using chinese healthcare datasets. In *Proceedings of the 34th Conference on Computational Linguistics and Speech Processing (ROCLING 2022)*, pages 356–362, 2022b.

B. Mohit. Named entity recognition. *Natural language processing of semitic languages*, pages 221–245, 2014.

D. Nadeau and S. Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.

T. Noraset, C. Liang, L. Birnbaum, and D. Downey. Definition modeling: Learning to define word embeddings in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Y. Qin, W. Yang, K. Wang, R. Huang, F. Tian, S. Ao, and Y. Chen. Entity relation extraction based on entity indicators. *Symmetry*, 13(4):539, 2021.

L. A. Ramshaw and M. P. Marcus. Text chunking using transformation-based learning. *Natural language processing using very large corpora*, pages 157–176, 1999.

S. Siami-Namini, N. Tavakoli, and A. S. Namin. The performance of lstm and bilstm in forecasting time series. In *2019 IEEE International conference on big data (Big Data)*, pages 3285–3292. IEEE, 2019.

C. J. Van Rijsbergen and W. B. Croft. Document clustering: An evaluation of some experiments with the cranfield 1400 collection. *Information Processing & Management*, 11(5-7):171–182, 1975.

Z. Wan, J. Xie, W. Zhang, and Z. Huang. Bilstm-crf chinese named entity recognition model with attention mechanism. In *Journal of Physics: Conference Series*, volume 1302, page 032056. IOP Publishing, 2019.

Y. Wu, M. Jiang, J. Lei, and H. Xu. Named entity recognition in chinese clinical text using deep neural network. *Studies in health technology and informatics*, 216:624, 2015.

T.-H. Yang, R.-C. Su, T.-E. Su, S.-S. Chong, and M.-H. Su. Scu-mesclab at rocling-2022 shared task: Named entity recognition using bert classifier. In *Proceedings of the 34th Conference on Computational Linguistics and Speech Processing (ROCLING 2022)*, pages 329–334, 2022.

Y. Yu, X. Si, C. Hu, and J. Zhang. A review of recurrent neural networks: Lstm cells and network architectures. *Neural computation*, 31(7):1235–1270, 2019.

N. Zhang, M. Chen, Z. Bi, X. Liang, L. Li, X. Shang, K. Yin, C. Tan, J. Xu, F. Huang, et al. Cblue: A chinese biomedical language understanding evaluation benchmark. *arXiv preprint arXiv:2106.08087*, 2021.

Q.-X. Zhang, T.-Y. Chi, T.-L. Yang, and J.-S. R. Jang. Crowner at rocling 2022 shared task: Ner using macbert and adversarial training. In *Proceedings of the 34th Conference on Computational Linguistics and Speech Processing (ROCLING 2022)*, pages 321–328, 2022.