Approach for Named Entity Recognition and Case Identification Implemented by ZuKyo-JA Sub-team at the NTCIR-16 Real-MedNLP Task

Koji Fujimoto†
Department of Real World Data Research and Development
Kyoto University
Kyoto, Japan
kfb@kuhp.kyoto-u.ac.jp

Mizuho Nishio
Department of Diagnostic Imaging and Nuclear Medicine
Kyoto University
Kyoto, Japan
nishiomizuho@gmail.com

Osamu Sugiyama
Department of Real World Data Research and Development
Kyoto University
Kyoto, Japan
sugiyama@kuhp.kyoto-u.ac.jp

Kana Ichikawa
Graduate School of Informatics
Kyoto University
Kyoto, Japan
ichikawa@kuhp.kyoto-u.ac.jp

Joseph Cornelius
IDSIA

Zurich, Switzerland
joseph.cornelius@idsia.ch

Oscar Lithgow-Serrano
IDSIA

Zurich, Switzerland
oscarwilliam.lithgow@idsia.ch

Vani Kanjirangat
IDSIA

Zurich, Switzerland
vanik@idsia.ch

Fabio Rinaldi
IDSIA

Zurich, Switzerland
fabio.rinaldi@idsia.ch

Aron Horvath
Department of quantitative Biomedicine
University of Zurich
Zurich, Switzerland
aronnorbert.horvath@uzh.ch

Farhad Nooralahzadeh
Department of quantitative Biomedicine
University of Zurich
Zurich, Switzerland
Farhad.Nooralahzadeh@uzh.ch

Michael Krauthammer
Department of quantitative Biomedicine
University of Zurich
Zurich, Switzerland
michael.krauthammer@uzh.ch

ABSTRACT

In this NTCIR-16 Real-MedNLP shared task paper, we present the methods of the ZuKyo-JA subteam for solving the Japanese part of Subtask1 and Subtask3 (Subtask1-CR-JA, Subtask1-RR-JA, Subtask3-RR-JA). Our solution is based on a sliding-window approach using a Japanese BERT pre-trained masked-language model., which was used as a common architecture for addressing the specific subtasks. We additionally present a method that makes extensive use of medical knowledge for the same case identification subtask3-RR-JA.

KEYWORDS

Medical Natural Language Processing, Named Entity Recognition, Case Reports, Radiology Reports, Case Identification, Lung Cancer, TNM Staging, Transformer, Data Augmentation.

TEAM NAME

ZuKyo

SUBTASKS

Subtask1-CR-JA, Subtask1-RR-JA, Subtask3-RR-JA (CI)

1 INTRODUCTION

It is not easy for researchers or companies outside of the hospital to have access to information stored in the electronic health records (EHR). For example, in Japan, the Next Generation Medical Infrastructure Act allows us to access medical information on an opt-out basis [1]. However, to the best of our knowledge, it is still very difficult to access texts and images in the EHR even under the Next Generation Medical Care Infrastructure Act. In order to promote research under this highly restrictive access policy to medical information, this competition (NTCIR-16) was held to establish a platform for analyzing EHR text using Natural Language Processing (NLP).

1.1 Team Overview

Our team consists of the JA sub-team and EN sub-team. Software programs of sub-teams were implemented based on weekly discussions between the JA and EN sub-teams. This paper describes the methods and results of the JA sub-team for the following three tasks: Subtask1-CR-JA (NER-CR-JA), Subtask1-RR-JA (NER-RR-JA), and Subtask3-RR-JA (CI-RR-JA). For an overview of the three tasks, please refer to the organizer's paper [2].

1.2 Case Report (CR)

As described above, strong access restrictions limit the use of raw EHR text in this competition. For this reason, open access Case Reports (CRs) submitted to conferences and journals were chosen as the target to analyze in this competition. As the organizer pointed out, the content of CRs is similar to that of EHR discharge summaries, where the process of a patient's diagnosis and treatment is summarized. Therefore, NLP for CRs is expected to be useful for also analyzing the EHR discharge summary.

1.3 Radiology Report (RR)

A RR represents a textual summary of findings in a medical image and is typically composed by a radiologist or a physician with training in this task. Most RRs in Japanese hospitals are written for CT and MRI examinations. In the Subtask1-RR-JA, and Subtask3-RR-JA (CI), RRs from lung cancer CTs are the targets to be analyzed. With the dominant research subject in radiology being image analysis, there is only limited experience with NLP of RRs, which makes this type of research challenging for radiology researchers.

1.4 Named Entity Recognition (NER) of CR and RR

NER has been studied intensively in past NLP studies. Therefore, the details of NER are omitted in this manuscript. Subtask1-CR-JA and Subtask1-RR-JA aim at extracting domain-specific sets of words from medical texts of CRs and RRs, respectively. Our implementation of NER is based on fine-tuning the BERT's Japanese pre-trained masked-language model [3,4]. Due to the small-sized dataset (N=148 for CR and N=72 for RR) in the NER tasks, we used data augmentation and increased the number of articles by a factor of 100.

1.5 Case Identification (CI) of RR

To create the dataset for the Subtask3-RR-JA (CI), multiple radiologists independently wrote RRs for the same sets of CT scans of several patients. Multiple RRs from the same patients are assigned the same group ID. Since the purpose of this task is to identify those RRs that describe the same CT case, it may be possible to use established methods for measuring document similarity, such as feature vectors extraction and comparison using Bag of Words techniques. However, the CI task presented us with unique challenges: For example, a Bag of Words approach may not work for this task, as the dataset consists only of RRs from lung cancer CT scans, where words are identical or quite similar.

1.6 Feature extraction with TNM staging for Subtask3-RR-JA (CI)

In order to group RRs without using NER, we used domain knowledge used in the diagnosis of lung cancer. Specifically, we applied knowledge of the TNM staging system of lung cancer to extract features from RRs. The TNM staging is the classification system to describe the progress of cancer, with the T factor indicating the extent of the primary tumor, the N factor the extent of lymph node metastasis, and the M factor the extent of metastasis to distant organs other than lymph nodes. Since all RRs in the Subtask3-RR-JA (CI) are diagnostic RRs for lung cancer, each RR contains sentences related to the TNM staging for lung cancer. Therefore, we decided to obtain the feature vector of the entire report by extracting features in terms of which sentences in RRs represent which TNM factors of lung cancer.

For readers not familiar with the TNM staging, a simplified version of T, N, and M factors of the Union for International Cancer Control (UICC) version 8 [4] is described below. In our implementation, we modified some parts of TNM factors to improve the extraction of feature vectors.

T factor (a simplified version)

- T1: Size of lung cancer, <3 cm
- T2: Size of lung cancer, 3-5 cm
- T3: (Size of lung cancer, 5-7 cm) or (Local invasion of chest wall, parietal pericardium, phrenic nerve)
- T4: (Size of lung cancer, >7 cm) or (Invasion to mediastinum, trachea, heart/great vessels, esophagus, vertebra, carina, recurrent laryngeal nerve)

N factor (a simplified version)

- N0: No regional lymph node metastasis
- N1: Metastasis in ipsilateral peribronchial and/or hilar lymph node and intrapulmonary node
- N2: Metastasis in ipsilateral mediastinal and/or subcarinal lymph nodes
- N3: Metastasis in contralateral mediastinal, contralateral hilar, ipsilateral, or contralateral scalene, or supraclavicular lymph node(s)

M factor (a simplified version)

- M0: No distant metastasis
- M1: Distant metastasis

As will be described later in detail, we estimated T, N, and M factors on a token/sub-token level with sentence-by-sentence augmentation of the RRs. The data augmentation, training step, ensemble of the multiple models are very similar to the methods we used for NER.

2 RELATED WORKS

2.1 NLP techniques used in our implementation

Recently, transformer-based NLP models have been used in various domains [3]. In our implementation, we used the Japanese BERT pre-trained masked-language model trained with the Japanese version of Wikipedia [4]. Since the dataset sizes for these three tasks were small, we expected that fine-tuning of the BERT pre-trained model would be effective for obtaining good performance in these three tasks. We used the cl-tohoku model as the pre-trained Japanese BERT models which are available in Transformers by Hugging Face(https://github.com/huggingface/transformers).

Since it was expected that using a fine-tuning approach (limiting the number of trainable layers) with BERT pre-trained models would not be sufficient for dealing with the limited number of datasets, data augmentation was also used in our implementation. In the past study, Easy Data Augmentation, a method for NLP data augmentation, was performed by randomly changing tokens in sentences [6]. Because Easy Data Augmentation is performed on token-level, Easy Data Augmentation was expected to deteriorate the performance in the NER and CI tasks. Therefore, we performed data augmentation on the NER-tag-level for subtask1 and on the sentence-level for the CI task.

2.2 Medical NLP for CR or EHR

There are several studies where medical NLP was used for analyzing CR or EHR. Peng et al compared three NLP tools for analyzing articles and abstracts of autism spectrum disorder obtained from PubMed [7]. Gurulingappa et al developed methods for the automatic extraction of drug-related adverse effects from CRs [8]. Based on their corpora, their method achieved a cross-validated F1 score of 0.70. Wang et al performed automatic classification of EHR into 7 types of infectious diseases using NLP [9]. They used an EHR of 20,620 patient cases covering 7 types of infectious diseases. Schulz et al constructed a new corpus comprising annotations for NER in CR obtained from PubMed Central's open-access library [10]. The corpus consists of 53 documents, which contain an average number of 156.1 sentences per document. Schulz et al compared four methods of NER and reported that Multi-Task Learning was the best method for NER of their corpus.

2.3 Medical NLP for RR

In radiology, content-based image retrieval systems for medical images have been studied intensively. For example, Müller et al summarize an overview of available literature in the field of content-based image retrieval systems for medical images [11]. On the other hand, compared to research with medical images, there are far less studies discussing NLP in radiology. Pons et al identified 67 relevant publications describing NLP methods that support practical applications in radiology for the following 5 categories: diagnostic surveillance, cohort building for epidemiologic studies, query-based case retrieval, quality assessment of radiologic practice, and clinical support services [12]. Although several NLP systems were developed for these 5 categories in radiology, Pons et al concluded that these systems were not actually used in routine clinical care or research. Recent NLP studies involving RR include the automatic classification of findings in head CT scans [13], automatic BI-RADS assessment in breast imaging [14], and the identification of abnormal findings in CT scans of children [15]. For Japanese RR, automatic classification of RR was investigated for eight diseases (Alzheimer's disease, lung cancer, myocardial infarction, fatty liver, disc herniation, medial collateral ligament injury, Elbow fracture, Achilles tendon injury) [16].

3 METHODS

In this section, we first describe the common and shared methods used for subtasks 1 and 3. After that, we will describe methods specific for the subtask3 (CI).

3.1 Labeling of the data for NER

The text data was first divided into tokens with MeCab, followed by sub-tokenization using WordPiece. The type of the entities provided from the organizer (i.e. <m>, <d>, <a>, etc.) was then added for each sub-tokens as the class label. The parts of the sentences without these entity labels were assigned a "other" label. This class label was further divided into two types. Specifically, if the token (or sub-token) was at the beginning of an entity, the flag to show the status (B-) was added to the label. If the token (or sub-token) was not at the beginning of an entity, this status flag (I-) was added to the token (label sets A). Label sets without discriminating (B-) and (I-) were also used for creating training dataset (label sets B). The examples of the class indices are shown in Tables 1 and 2. In our approach, estimation of the tag attributes was treated independently of the estimation of the entity class. In this way, we could use the same framework for estimating the tag attributes. Specifically, certainties (positive, suspicions, negative, general, correction) were assigned indices of 0,1,2,3,4 and states (scheduled, executed, negated) were assigned indices of 6,7,8.

Table 1. Example of class indices in label sets A and B.

Class index	Label sets A	Label sets B
0	B-disease	disease
1	I-disease	
2	B-timex3_date	timex3_date
3	I-timex3_date	
4	B-timex3_time	timex3_time
5	I-timex3_time	
6	B-timex3_dur	timex3_dur
7	I-timex3_dur	
36	B-other	other
37	I-other	

3.2 Our dataset format for NER

An example of our dataset format is shown in Table 3. The articles were processed by MeCab, which split the articles' plain text into tokens. These tokens were further divided into sub-tokens using WordPiece. With N={3,4,6,8,10,15,30,60}, a sentence consisting of N*2+1 sub-tokens was taken from each article in a sliding window manner, with the restriction that the

beginning of each sentence should be the beginning of the token (but not sub-token). For each sentence, class ID for the N+1th sub-token was the target for the BERT estimate (2nd column in Table 3).

Table 2. Example of class indices in Certainties and States.

Class index	Certainties	States
0	positive	
1	suspicious	
2	negative	
3	general	
4	correction	
5		scheduled
6		executed
7		negated

Table 3. Example of the dataset with N=3. The target sub-tokens (4th sub-token for this case) are shown in the bracket [].

Sentence	Class index / label
散見し, [明らか] な肺癌	28 / B_feat
し,明らか [な] 肺癌の	36 / B_other
,明らかな [肺] 癌の肝	0 / B_dis
明らかな肺 [癌] の肝浸	1 / I_dis
な肺癌 [の] 肝浸潤	1 / I_dis

3.3 Our model

To perform NER, we chose to use BERT-based topic modeling. Specifically, we fine-tuned BERT's Japanese pre-trained masked-language model to estimate class of the entity for the N+1-th sub-token from the sentence which consists of N*2+1 sub-tokens. Input was a train of sub-tokens and the output to be trained was the class label [CLS]. The first column of the output of the pre-trained BERT model was further put into the dropout layers and the fully-connected layer. The number of features for this fully-connected layer was set to 41, such that all the classes we prepared for NER have the unique class ID.

3.4 Data augmentation

We consider that elements belonging to the same entity are exchangeable for creating augmented dataset for the NER task. Therefore, we first built a dictionary for each entity, and the augmented dataset was built based on the algorithm 1 shown below. This process was repeated 100 times for each article, resulting in 14800 (augmented) articles for CR-JA and 7200 articles for RR-JA.

3.5 Parameters for fine-tuning the model

We used the following parameters to fine-tune the BERT pre-trained model.

- N: the number of sub-tokens before and after the target sub-token.
- E: number or epochs
- LR: learning rate
- B: batch size
- DO: probability for the dropout layer
- V: if V>0, the number of learnable layers (at the bottom) in the pretrained BERT model.
- A, B: the type of the label sets used to build the training dataset.

Algorithm 1. Algorithm for the data augmentation.

1. Choose a part of a sentence
2. If the part is not tagged as an entity, use the original sentence part.
3. if the part is tagged as an entity,
4. If the entity E is either one of the class C={d, a, timex3, t-test, t-key, t-val, m-key, m-val}, randomly choose an element from the dictionary for the entity E.
5. Continue until the end of the article.

3.6 Model training for NER

We split the dataset for a 5-fold CV for the model training. However, due to the limited amount of time, we only performed fold 0 of 5-fold CV (118 articles for RR and 57 articles for CR). Models were trained with varying parameters on a grid-search basis. Due to the limited amount of time, only part of the combination was completed for training. For final model training (used for ensemble inference), we used all the data with NER tags (148 articles for CR and 72 articles for RR).

3.7 Inference of entity (NER tag) with ensemble

For each sub-token, outputs of the several models after the softmax function were averaged to estimate a class label. The output after the ensemble was a 41-dimensional vector of probabilities for each sub-token.

3.8 Inference of entity attributes

We independently trained models to estimate the attributes (separately for certainties and states). The methods (parameters for fine-tuning the model, model training, inference (ensemble)) used were the same as the methods for NER. The outputs for certainties and states were finally combined with the estimates for the entities. An estimates for an attribute was only accepted when the corresponding entity had an attribute in the first place. Otherwise, the output of the model for the token attribute was ignored.

3.9 Labeling of the data for CI

We developed a dedicated method for CI. We did not use the tags and the attributes in the training dataset provided by the organizer. Instead, we built a sentence-wise tag dedicated to the radiology reports for TNM staging of lung cancer. The representative tag types used for this are shown in Table 4. Here, we used an approximation that each sentence in RR refers to only one of the classes we defined. Although descriptions of the TNM staging in RRs are not always correct, we decided to ignore such imperfections for this competition. In other words, we assumed that TNM factors were correctly described in all RRs. It was also possible that different patients had the same TNM factors. In order to deal with this problem, we modified some parts of the TNM factors. Specifically, metastases were

further split into bone and other metastases. The labeling for the training was performed by one of the authors with an experience of over 20 years in diagnostic radiology.

Table 4. Example of class indices used for the CI task.

Class index	Customized tag of TNM
4	B-DistantMeta
10	B-N0
12	B-N2
14	B-N3
16	B-T1
18	B-T2
24	B-T3
39	B-T4

3.10 Model training for CI

The method for model training was similar to the method used for NER except that we chose models with a weighted F-measure >0.6 for the final ensemble-based inference.

3.11 Estimation of TNM staging and CI

The output of the ensemble of the trained models was first summarized as the number of tokens for each class for each report, and then the reports were summarized for T factor, N factor, and M factor by using an algorithm shown in Algorithm 2.

Algorithm 2. Algorithm for assigning one of the T, N, M factors from the output of the model for the article. The number of tokens for X is expressed as N(X). T(or N or M) factor for the article is expressed as f(T) or f(N) or f(M)

For T factor

1. if N(T1)=N(T2)=N(T3)=N(T4)=0, f(T) was left blank
2. if N(T3) > 10, f(T)=T3
   if N(T3) > 0 and N(T2)=N(T1)=0, f(T)=T3
3. else
   a. If N(T1) > 10, f(T)=T1
   b. else
      i. if N(T2) > N(T1), the f(T)=T2
      ii. else the T factor for the article was set to T1

For N factor

4. if N(N2) + N(N3) < 10, f(N)=N0
5. else
   a. if N(N3) > 9, f(N)=N3
   b. otherwise f(N)=N2

For M factor

6. if N(BM1) + N(DistantMeta)==0, f(M)=M0
7. else f(M)=M1

The method for CI from the estimated TNM staging (based on the number of tokens) is summarized in Algorithm 3.

Algorithm 3. Procedure for assigning case number based on the estimated TNM staging.

- T3N3M1 -> case 1
- T2or1N3M1 -> case2
- T3N3orBlankM0 -> case 3
- T2N3M0 -> case4
- T1orBlankN0Mx -> case5
- T2N0Mx -> case6
- otherwise -> case7 or case2

Based on the information provided by the organizer, the test dataset for CI (63 radiology reports) should be divided into 7 cases, each of which consists of 9 reports. However, we did not apply this knowledge to restrict the number of reports assigned to the same cases.

4 RESULTS

4.1 Results calculated by the organizers

We submitted results of four different models for Subtask1-CR-JA, four results for Subtask1-RR-JA, and one result for Subtask3-RR-JA (CI). For Subtask1-CR-JA and Subtask1-RR-JA, the scores for the best model by the moderator's criteria are described here. For Subtask1-CR-JA,, entityP, entityR, and entityF of all entity-level targets are 35.55, 35.74, and 35.65, respectively and jointP, jointR, and jointF of all joint-level targets are 28.82, 30.04, and 29.93, respectively. For Subtask1-RR-JA, entityP, entityR, and entityF of all entity-level targets are 55.42, 65.64, and 60.10, respectively and jointP, jointR, and jointF of all joint-level targets are 40.16, 47.56, and 43.55, respectively. The F-measures of our best results for Subtask1-CR-JA, and Subtask1-RR-JA, are shown in Tables 5-8. For Subtask3-RR-JA (CI), our score of Normalized Mutual Information was 0.4161 (* we found a bug in our code after the task deadline, and the number of tokens for stage T4 were all zero in our original submission. After fixing the bug and updating algorithms 2 and 3, the updated inference was scored as 0.4622).

Table 5. F-measures for Subtask1-CR-JA in the tag-level

Type of Named Entity	F-measure
All targets	35.65
<a>	30.61
<d>	45.41
<m-key>	33.75
<m-val>	20.34
<t-key>	13.86
<t-test>	28.23
<t-val>	13.12
<timex3>	48.30

Table 6. F-measures for Subtask1-CR-JA in the joint-level

Pair of Named Entity and Attribute	F-measure
All targets	29.93
<d> + general	6.35
<d> + negative	8.33
<d> + positive	37.51
<d> + suspicious	0.00
<m-key> + executed	26.25
<m-key> + negated	28.57
<m-key> + other	0.00
<m-key> + scheduled	0.00
<t-test> + executed	29.65
<t-test> + other	0.00
<timex3> + age	68.90
<timex3> + date	46.42
<timex3> + duration	0.00
<timex3> + med	47.06
<timex3> + misc	2.00
<timex3> + set	25.64
<timex3> + time	6.45

Table 7. F-measures for Subtask1-RR-JA in the tag-level

Type of Named Entity	F-measure
All targets	60.10
<a>	62.71
<d>	59.01
<t-test>	44.16
<timex3>	91.43

Table 8. F-measures for Subtask1-RR-JA in the joint-level

Pair of Named Entity and Attribute	F-measure
All targets	43.55
<d> + positive	41.15
<d> + suspicious	12.24
<d> + negative	2.56
<d> + general	0.00
<t-test> + executed	40.00
<t-test> + negated	0.00
<t-test> + other	0.00
<timex3> + date	9.33
<timex3> + duration	0.00
<timex3> + med	80.00

4.2 Dataset size after data augmentation

After the data augmentation, the number of articles for Subtask1-CR-JA was 11800 for the parameter optimization session, and 14800 for the final training session. Similarly, the number of articles for Subtask1-RR-JA and Subtask3-RR-JA (CI) was 5700 for the parameter optimization, and 7200 for the final training session. The number of lines after building the training dataset was approximately 1605K (N=60) to 1940K (N=3) for Subtask1-CR-JA, 362K (N=30) to 436K (N=3) for Subtask1-RR-JA, 157K (N =60) to 201K (N =3) for Subtask3-RR-JA (CI).

4.3 Models used for the ensemble (Subtask1-CR-JA, Subtask1-RR-JA, Subtask3-RR-JA (CI))

Of the four submitted models, we describe here the model with the best score according to the organizer's evaluation. Our best model for the Subtask1-CR-JA was an ensemble of 35 models. As mentioned in the methods section, we used two types of label sets for training the model. Our best model was the ensemble of the models trained with label sets A (N=17) and label sets B (N=18). For the entity attributes (both for certainties and states), an ensemble of 7-8 models was used for the final inference (7 for states, and 8 for certainties). Similarly, for Subtask1-RR-JA, an ensemble of 15 models was used for the entity labels, and an ensemble of 8 models was used for entity attributes. As mentioned in the methods, the threshold of weighted F-measure >0.6 was used to select models for the CI task. The model that met this criterion (at the time of submission) was 11.

4.4 Sub-token level F-measures for Subtask1-CR-JA with label set A

To compare the performance of the models with different training parameters, the sub-token level F-measures for the validation dataset in our fold-0 data are summarized in Table 9.

4.5 Results of the case identification

By applying the algorithm shown in Algorithms 2 and 3, the estimated number of reports for each case ranged between 6 to 16. As shown, the official score of our submitted data was scored by the normalized mutual information. Our score was 0.4161 (updated score, 0.4622).

5 DISCUSSION

5.1 NER

As shown in Figure 1, the sub-token level F-measures tend to be better with a larger number of N (the number of tokens before and after the target token) for models with no trainable layers in BERT (v=0). On the other hand, models with N=3 and 12 trainable layers (v=12) performed better than models with v=0. However, for entities such as the disease and the anatomy, models with a larger N(=30) at epoch 3 (brown line in Figure 1) performed better than the models with v=12 (green and red lines in Figure 1). The improvement of the models over the

Figure 1. The summary of the representative models for Subtask1-CR-JA is shown. The sub-token level F-measures for each entity class for the validation dataset in our fold-0 data for the representative models with different training parameters are shown. The number of supports (i.e. the number of tokens in the validation dataset) is shown in blue bars. The models with twelve trainable layers (v12) are shown in green and red lines. The remaining models are trained with v=0. Batch size=32, Dropout=0.2. Learning rate (LR)=0.00004 was used for v12 and LR=0.00002 for v0. Abbreviations: n=number of tokens before and after the target token, e=number of epochs, b=batch size, do=dropout, v=number of learnable layers in the pre-trained BERT model.

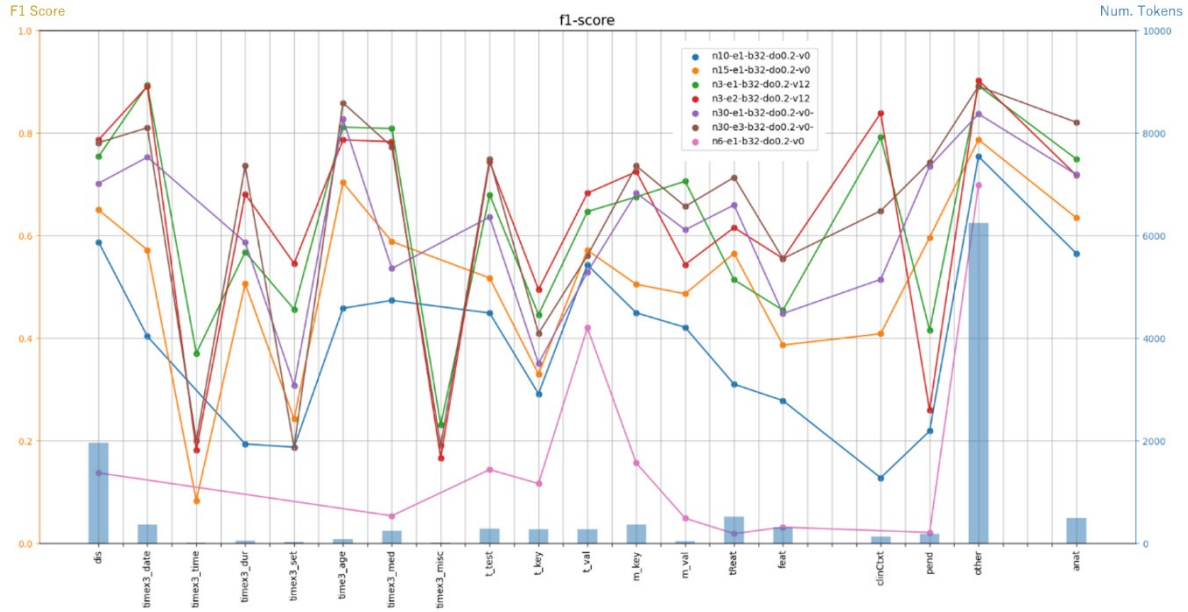
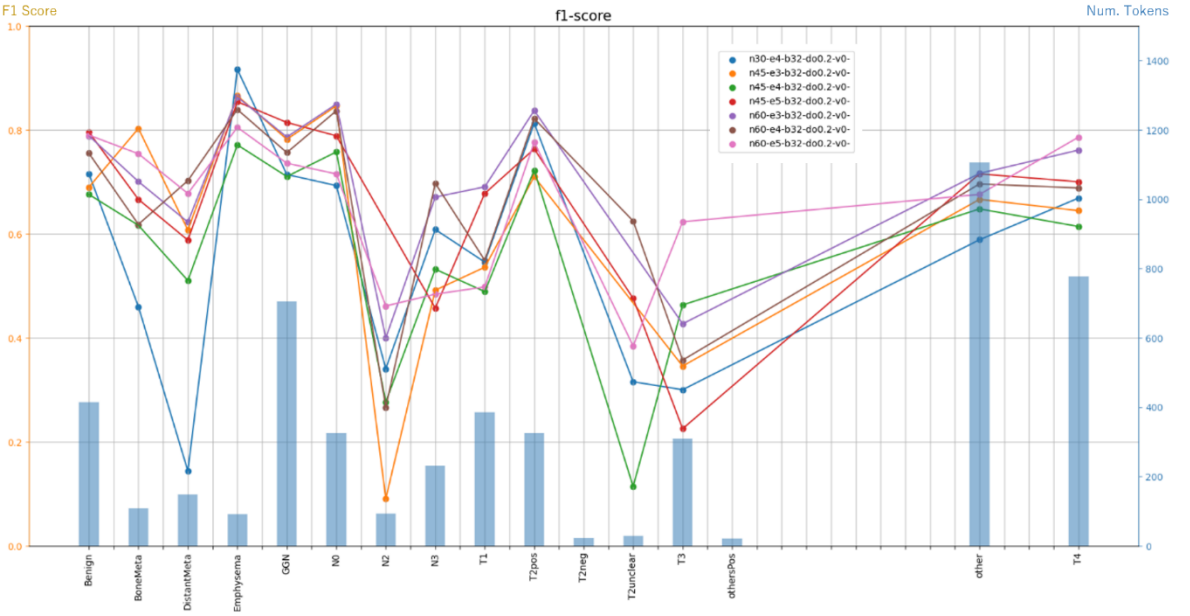


Figure 2. The summary of the representative models used for Subtask3-RR-JA (CI) is shown. The sub-token level F-measures for each entity class for the validation dataset in our fold-0 data for the representative models with different training parameters are shown. The number of supports (i.e. the number of tokens in the validation dataset) is shown in blue bars. All the models shown here are trained with v=0. Batch size=32, Dropout=0.2, learning rate (LR)=0.00002. Abbreviations: n=number of tokens before and after the target token, e=number of epochs, b=batch size, do=dropout, v=number of learnable layers in the pre-trained BERT model.



epoch varied for the target entity classes. The reason for the use of a small N for models with v=12 was to avoid overfitting. However, the number of models who finished training at the deadline of the competition was not sufficient to analyze this aspect. It should be noted that the models with a smaller N and larger v(=12) performed similarly (with some difference) to the models with a large N and smaller v(=0). This difference was advantageous for our ensemble inference strategy. We were not able to choose the model used for the inference based on the model performance. Instead, we relied on the ensemble strategy itself. Ideally, the best model should be chosen for each N.

5.2 CI

As described in the organizers' overview paper, the CI task of RRs does not represent a likely clinical situations. Therefore, it

is quite difficult to find previous studies discussing a related task. However, the CI task can be seen as equivalent to a similar sentence retrieval task, for which much prior work exists. Nevertheless, we could not find previous studies where TNM staging was used for similar sentence retrieval of RRs. On the other hand, there are several studies where information about TNM staging was extracted from free text of RRs. Nobel et al developed and evaluated their extractor of the T factor from RRs of lung cancer patients [17]. The accuracy of their extractor of the T factor was 0.89. Gupta et al developed an NLP system for extracting TNM factors, and the accuracy of their system was up to 59% for the T factor, 36% for the N factor, and 41% for the M factor [18]. Hu et al extracted information about the TNM staging of lung cancer and lung-cancer-related findings from Chinese free text of RRs [19]. Their BERT-based system consisted of three modules (NER, relation classification, and post-processing modules), which achieved a macro-F1 score of 94.57% and a micro-F1 score of 96.74% for all the 22 questions related to the TNM staging.

In this competition, the dataset for the CI task is the same as for subtask1, in which annotations for NER are available. If we can correctly estimate the <a>-, <d>-, and <f>-tag of NER (including attributes) in the test set of this task, we might be able to group RRs effectively. However, this approach depends on the performance of the NER subtask. In addition, relationships between the entities should also be correctly estimated. Therefore, we decided not to use the NER-based method for grouping RRs. Instead, we decided to fully exploit the fact that the document (radiology report) aims at describing TNM staging, which we used as the training label. Figure 2 shows the performance of the representative models trained for this task. Although the validity of adding a “tag” to each sentence is debatable, we found that this type of annotation is learnable. The annotation effort of our method is low because only one label is required for each sentence (as comparted to each word or phrase). However, this method requires knowledge about the TNM staging system. The models with epochs 4 (green and brown) tend to show a lower performance compared to other models with the same number of epochs (orange and purple, respectively). This could be due to overfitting indicating that the augmentation strategy was not very effective compared to the one used for the NER task.

6 CONCLUSION

In this manuscript, we have described our approach for the three tasks: Subtask1-CR-JA (NER-CR-JA), Subtask1-RR-JA (NER-RR-JA), and Subtask3-RR-JA (CI-RR-JA) based on a sliding-window approach using Japanese BERT pre-trained masked-language model. A lot of methods used for these subtasks are shared, regardless of the task differences. We also discuss a method that makes extensive use of medical knowledge for the same case identification subtask3-RR-JA.

Contribution

KF and MN performed most of the experiments described in this paper. All authors participated in several discussions and exchanged ideas. All authors read the paper and approved it.

REFERENCES

[1] Next Generation Medical Infrastructure Act | e-Gov. https://elaws.e-gov.go.jp/document?lawid=429AC0000000028 (accessed Feb. 26, 2022).

[2] Shuntaro Yada, Yuta Nakamura, Shoko Wakamiya, and Eiji Aramaki. 2022. Real-MedNLP: Overview of REAL document-based MEDical Natural Language. Proceedings of the NTCIR-16 Conference on Evaluation of Information Access Technologies.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics. 4171–4186.

[4] GitHub - cl-tohoku/bert-japanese: BERT models for Japanese text. https://github.com/cl-tohoku/bert-japanese (accessed Feb. 23, 2022).

[5] Wanyin Lim, Carole A. Ridge, Andrew G. Nicholson, and Saeed Mirsadraee. 2018. The 8th lung cancer TNM classification and clinical staging system: review of the changes and clinical implications. Quantitative Imaging in Medicine and Surgery. 8, 7 (August 2018), 709–718.

[6] Jason Wei and Kai Zou. 2019. EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. Proceedings of EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing. 6382–6388.

[7] Jacqueline Peng, Mengge Zhao, James Havrilla, Cong Liu, Chunhua Weng, Whitney Guthrie, Robert Schultz, Kai Wang, and Yunyun Zhou. 2020. Natural language processing (NLP) tools in extracting biomedical concepts from research articles: a case study on autism spectrum disorder. BMC Medical Informatics and Decision Making. 20, 11 (2020), 322.

[8] Harsha Gurulingappa, Abdul Mateen Rajput, Angus Roberts, Juliane Fluck, Martin Hofmann-Apitius, and Luca Toldo. 2012. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. Journal of Biomedical Informatics. 45, 5 (October 2012), 885–892.

[9] Mengying Wang, Zhenhao Wei, Mo Jia, Lianzhong Chen, and Hong Ji. 2022. Deep learning model for multi-classification of infectious diseases from unstructured electronic medical records. BMC Medical Informatics and Decision Making. 22, 1 (2022), 41.

[10] Sarah Schulz, Jurica Ševa, Samuel Rodriguez, Malte Ostendorff, and Georg Rehm. 2020. Named Entities in Medical Case Reports: Corpus and Experiments. Proceedings of the 12th Conference on Language Resources and Evaluation. (2020), 4495–4500.

[11] Henning Müller, Nicolas Michoux, David Bandon, and Antoine Geissbuhler. 2004. A review of content-based image retrieval systems in medical applications–clinical benefits and future directions. International Journal of Medical Informatics. 73, 1 (February 2004), 1–23.

[12] Ewoud Pons, Loes M.M. Braun, M. G.Myriam Hunink, and Jan A. Kors. 2016. Natural language processing in radiology: A systematic review. Radiology. 279, 2 (May 2016), 329–343.

[13] John Zech, Margaret Pain, Joseph Titano, Marcus Badgeley, Javin Schefflein, Andres Su, Anthony Costa, Joshua Bederson, Joseph Lehar, and Eric Karl Oermann. 2018. Natural language-based machine learning models for the annotation of clinical radiology reports. Radiology. 287, 2 (May 2018), 570–580.

[14] Sergio M. Castro, Eugene Tseytlin, Olga Medvedeva, Kevin Mitchell, Shyam Visweswaran, Tanja Bekhuis, and Rebecca S. Jacobson. 2017. Automated annotation and classification of BI-RADS assessment from radiology reports. Journal of Biomedical Informatics. 69, (May 2017), 177–187.

[15] Fredrik A. Dahl, Taraka Rama, Petter Hurlen, Pål H. Brekke, Haldor Husby, Tore Gundersen, Øystein Nytrø, and Lilja Øvrelid. 2021. Neural classification of Norwegian radiology reports: using NLP to detect findings in CT-scans of children. BMC Medical Informatics and Decision Making. 21, 1 (March 2021), 84.

[16] Taro Tada and Kazuhide Yamamoto. 2019. Effect of Preprocessing for Distributed Representations: Case Study of Japanese Radiology Reports. Proceedings of the 2019 International Conference on Asian Language Processing, IALP 2019. (November 2019), 29–34.

[17] Martijn Nobel, Sander Puts, Frans C.H. Bakers, Simon G.F. Robben, and André L.A.J. Dekker. 2020. Natural Language Processing in Dutch Free Text Radiology Reports: Challenges in a Small Language Area Staging Pulmonary Oncology. Journal of Digital Imaging. 33, 4 (August 2020), 1002–1008.

[18] Khushbu Gupta, Ratchainant Thammasudjarit, and Ammarin Thakkinstian. 2019. NLP Automation to Read Radiological Reports to Detect the Stage of Cancer Among Lung Cancer Patients. Proceedings of the 2019 Workshop on Widening NLP, 138–141.

[19] Danqing Hu, Huanyao Zhang, Shaolei Li, Yuhong Wang, Nan Wu, and Xudong Lu. 2021. Automatic Extraction of Lung Cancer Staging Information From Computed Tomography Reports: Deep Learning Approach. JMIR Medical Informatics. 9, 7 (2021), e27955.