Efficient rule-based approaches for tagging named entities and relations in clinical text

Dongmin Kim[1], Soo-Yong Shin[2], Hee-Woong Lim[3] and Sun Kim[4]
[1]Hayoom Research, MD, USA
[2]Department of Digital Health, SAIHST, Sungkyunkwan University, Seoul, South Korea
[3]Institute for Diabetes, Obesity and Metabolism, Perelman School of Medicine, University of Pennsylvania, PA, USA
[4]National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, MD, USA
dmkim@hayoom.com, sy.shin@skku.edu, heewlim@pennmedicine.upenn.edu, sun.kim@nih.gov

***Abstract*—We present a rule-based system that efficiently identifies named entities and their relational tuples from clinical narrative text. For named entity recognition (NER), we build a dictionary for each named entity, namely, family member, living status and observation. While dictionaries are essentially extracted from the BioCreative/OHNLP training data, we also extend the observation dictionary by crawling the Mayo disease-condition index website. This is particularly to detect new concepts appearing in the BioCreative/OHNLP test set. The entities recognized by our dictionaries are then fed into the relation extraction (RE) module, where we segment each sentence and aggregate them back to produce multi-sentence paragraphs. Each paragraph is coherent with respect to an identified family member, i.e. it always contains one family member and any observation and/or the description of one's living status associated with the given family member. Once paragraphs are identified, it is straightforward to produce the final relation candidates by pairing entities in each paragraph. Applied to the BioCreative/OHNLP training set, our system provides 0.83 and 0.60 F1 scores for NER and RE subtasks, respectively.**

Keywords—named entity recognition; relation extraction; rule-based approaches; clinical notes

I. Introduction

Electronic health record (EHR) systems have been widely adopted in hospitals and medical facilities. However, much clinical data is written in narrative text. To extract necessary information from clinical text, natural language processing (NLP) techniques are often applied to EHR (1-3). Although there is some progress, clinical NLP methods should be further improved for practical usage. Recently, deep learning approaches have demonstrated their power in analyzing narrative clinical text (4), but still traditional NLP approaches should be used to analyze narrative clinical text better.

The traditional NLP methods mostly rely on part-of-speech tagging and parsing in analyzing text. However, frequent use of synonyms and incomplete sentences in EHRs make it difficult to apply NLP approaches successfully. Therefore, we here propose rule-based NLP methods for named entity recognition (NER) and relation extraction (RE) in clinical notes. To demonstrate the performance of our proposed methods, we participated in the family history information extraction task of the 2018 BioCreative/OHNLP Challenge. The task is to extract family disease history from unstructured clinical notes and it consists of two subtasks: 1) entity (family members and disease names) identification and 2) extraction of family members and corresponding observations (5). In the following sections, we describe our approaches and our performance on the BioCreative/OHNLP dataset.

II. System Overview

Following the challenge structure, we designed a system with two major components. Given a textual input, the system first performs named entity recognition, and then extracts appropriate relations among them.

A. Named Entity Recognition

Named entity recognition is the task to locate and classify a word or a chunk of words into pre-defined categories. For this part of the challenge, the system should identify two types of named entities, "FamilyMember" and "Observation." "FamilyMember" encompasses names of family members such as father, mother, brother, etc. In addition, the system may identify the indication of side of a family, e.g. paternal or maternal when available. Table I shows the complete list of family members that the system identifies. Note that the first degree relatives do not allow the side of family information to be associated.

TABLE I. FamilyMember entities

Without family side	With family side
Father, Mother, Parent, Sister, Brother, Daughter, Son, Child, Sibling	Grandmother, Grandfather, Grandparent, Cousin, Aunt, Uncle

While "FamilyMember" only includes a handful of possible tokens and their combinations, "Observation" is defined to have a much larger set of named entities, that is, names of known diseases and their descriptive qualifiers. Table II lists a few examples of the "Observation" entity type:

TABLE II. OBSERVATION ENTITIES

Ciprofloxacin and Dexamethasone, Atazanavir and Cobicistat, Tirofiban, Otic Route, Nosebleeds, Pongidae disease

In order to establish a baseline performance, we built a classification-based NER system using the NLTK toolkit (6) (See III.A for more details). After a careful review of the baseline performance, we concluded that a *simpler* approach would outperform the baseline. Here, we provide the rationale behind our decision.

- "FamilyMember" has a very small number of possible tokens (17 to be exact), with a pattern that can be easily standardized when the side of family information is present.
- "Observation" contains names of diseases, which means, in turn, they are already known precisely. In other words, we can build a list or dictionary of possible token(s) prior to performing NER.
- Disease names are somewhat unique, i.e. they can be recognized by the corresponding token(s) with high accuracy.
- The training corpus is limited in terms of its size, more sophisticated methods would overfit too easily.

In our case, token location or order information, which often plays a crucial role in general domain NER tasks, does not seem as critical. Furthermore, in order to handle multi-word named entities, we should formulate the NER task as multi-label classification. However, this may introduce unnecessary complexity and further degrades the overall performance of the baseline.

Based on our observation above, we design our NER component to use a pre-built dictionary of each named entity, and combine them using pattern matching. Although this approach is straightforward, its limitation is also very clear; the coverage of each named entity dictionary dictates the accuracy of the results. This is especially troublesome for "Observation", as the limited availability of annotated texts inherently entails the limited coverage of the dictionary. To alleviate this issue, we extend the "Observation" dictionary with an external resource. Specifically, we crawl and extract disease, symptom, and drug names from the Mayo website (7), and supplement the dictionary built upon the BioCreative/OHNLP training set.

B. *LivingStatus and Its Score*

Before we describe our relation extraction component, we note that there is an additional entity type; "LivingStatus" and living status scores. Although it is not a part of the NER subtask, they are required to be recognized prior to the RE subtask. Therefore, we also identify "LivingStatus" entities in our NER component in addition to "FamilyMember" and "Observation." We also note that "FamilyMember" and "Observation" entities are commonly associated with nouns, however, "LivingStatus" is mostly identified with adjectives. Table III shows a few examples. For brevity, we skip the details of scoring. Please refer to the task overview (5) for more details.

TABLE III. LIVINGSTATUS ENTITIES

LivingStatus	Score
alive and well	4
alive	2
death	0

C. *Relation Extraction*

Similar to NER, relation extraction is another common task in NLP, that normally follows NER. In this part of the challenge, we extract two types of relations (Table IV). For reference purposes, we label each relation as "R1" and "R2", respectively.

TABLE IV. FAMILY HISTORY RELATIONS

Named entities	Type
FamilyMember, side of family, LivingStatus, living status score	R1
FamilyMember, side of family, Observation	R2

The relations in the BioCreative/OHNLP training set often take the form, "subject-verb-object", and we find a few noticeable characteristics as follows.

- We need to compute a numerical score for "LivingSatus" entities in R1.
- A "FamilyMember" entity can belong to multiple relations.
- "FamilyMember" can be in either multiple relations of the same type, or even in different types.
- Multiple relations can be produced from a single sentence.
- A single relation can span over multiple sentences.

To investigate the degree of additional complexity (or lack of it), we recast the problem as a binary classification task, where each token is tagged based on its membership to a relation. We then build another baseline tagging model to see how it performs (see III.B for more details). Although the overall accuracy marks over 0.8 for the 20-fold cross-validation experiment, the performance in each fold fluctuates significantly, ranging from 0.3 to 0.9. We suspect it is yet another sign of a small dataset, which indicates the RE subtask is more challenging than the NER problem. To that end, instead of building a generic relation extractor, we decided to focus on exploiting the specifics of the given task.

First, we observe that many of sentences include one "FamilyMember" entity, and when there exists any "Observation" and/or "LivingStatus", their simple combinations constitute relations. Secondly, when a sentence contains

multiple "FamilyMember" entities, it produces either a relation for each "FamilyMember" with common "Observation" and/or "LivingStatus", or a relation for each "FamilyMember" and the corresponding other entities. Table V shows our *split* processing strategies for sample sentences.

TABLE V. SPLIT PROCESSING OF SENTENCES

	Case 1	**Case 2**
Sample Sentence	James and his brother had fever.	James had fever while his brother was healthy.
Split Rule	If no other entities found between "FamilyMember" entities, duplicate the found entities.	Otherwise, split at the conjunctions or connecting words of each "FamilyMember."
Derived Sentences	* James had fever. * And his brother had fever.	* James had fever. * While his brother was healthy.
Sample Relations	<James, …, fever> <Brother, …, fever>	<James, …, fever> <Brother, …, healthy>

Finally, for a relation that spans over multiple sentences, we aggregate candidate sentences into a *paragraph*, treat it as a single sentence, and then invoke the first or second case processing as we discussed above. Currently, we simply combine subsequent sentences to the previous ones, whenever we find pronouns at the beginning of new sentence. At this point, these cases are by no means fully inclusive, there are many different cases and sub-cases we will need to investigate to improve the system performance.

III. IMPLEMENTATION AND EXPERIMENTS

We implemented our system and performed experiments in Python. The baseline models were built using the open-source packages such as NLTK (6) and Stanford CoreNLP (8).

A. Baseline Performance

1) Named Entity Recognition

To establish the baseline NER, we first parse the annotated XML files and the corresponding texts. Based on the parsed information, we create an IOB (Inside, Outside and Begin) format file (6) to train a classification-based named entity tagger. In particular, we use gradient boosting (9) for the baseline tagging model. Fig. 1. shows the confusion matrix of the baseline performance based on 20-fold cross-validation. We use more folds than usual (e.g. 10-fold) due to the small number of training examples. For demonstration purposes, we included all the available entities in the training corpus. Note that only the labels, (B-/I-) (FamilyMember | LivingStatus | Observation) are of our interest, and their accuracies are in the 0.6 to 0.8 range.

2) Relation Extraction

Similarly, we parse out relations from the training corpus to create an IOB-format file. Unlike the NER task, the number of labels (i.e. relations) is unknown, hence we formulate the relation extraction task as a binary (*in* or *out-of* relation) tagging problem. Fig. 2 shows the confusion matrix of the baseline performance based on 20-fold cross-validation. Note that the accuracy marks over 0.7.

Although the baseline performance for each subtask is encouraging, the problem persists throughout the experiments: the performance of individual runs within the cross-validation fluctuates vastly with large variance; it renders the system built on top of such underlying algorithms highly unreliable.

B. Our System Performance

As explained earlier, the observations from the baseline experiments lead us to investigate a simpler model. Table VI shows the performance of our system on the BioCreative/OHNLP training set. The scores are computed by the evaluation script provided by the task organizers.

TABLE VI. THE SYSTEM PERFORMANCE ON THE TRAINING SET

	NER	**RE**
True Positive	1400	681
False Positive	370	487
False Negative	188	427
Precision	0.7910	0.5830
Recall	0.8816	0.6146
F1	0.8338	0.5984

IV. DISCUSSION

We built a simple but efficient rule-based system to identify named entities and to extract relations from clinical notes. Although the challenge simplifies a few conditions such as negation information, it still poses difficulties, especially from the limited size of available annotated data. It would be interesting to use the developed system to annotate (or speed up the annotation process) to increase the size of the corpus. Though it would be challenging, we believe that a powerful system could be built by i) properly designing and specifying the structure of the dataset and by ii) building a sizable data set to train more sophisticated methods.

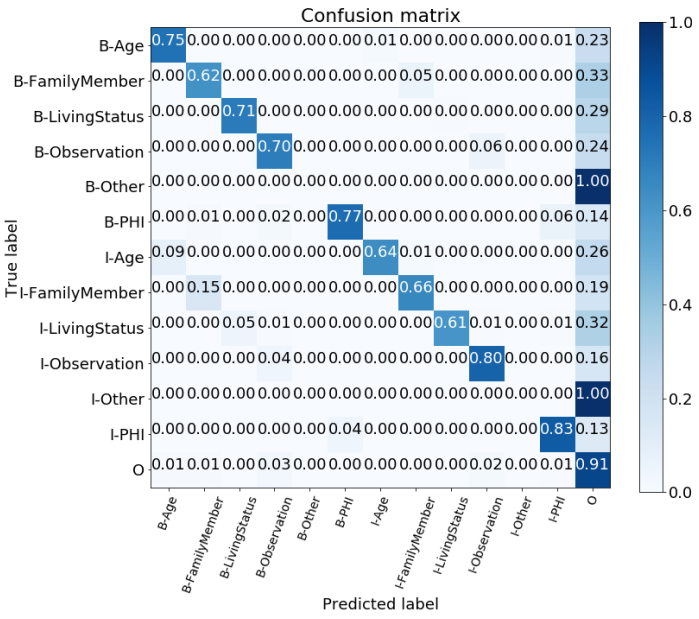

FIGURE 1. CONFUSION MATRIX OF THE BASELINE NER SYSTEM. FOR INVESTIGATIVE PURPOSE, ALL THE AVAILABLE ENTITIES ARE TAGGED (AGE, FAMILYMEMBER, LIVINGSTATUS, OBSERVATION, PHI, AND OTHER). THE TOKENS OUTSIDE OF NAMED ENTITIES ARE ASSIGNED TO THE "O" CLASS. FOR MULTI-WORD NAMED ENTITIES, THE BEGINNING OF AN ENTITY IS TAGGED WITH THE "B-" PREFIX WHILE THE FOLLOWING TOKENS ARE MARKED WITH THE "I-" PREFIX.

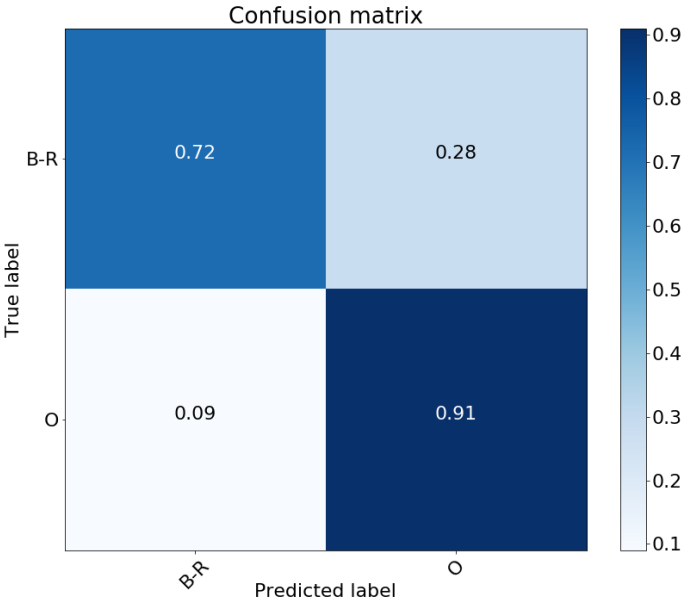

FIGURE 2. CONFUSION MATRIX OF THE SIMPLIFIED RELATIONAL ENTITY CLASSIFICATION. THE TAG "B-R" INDICATES THE TOKEN IS PARTICIPATING IN A RELATION WHILE "O" IS ASSIGNED TO OUT-OF-RELATION TOKENS.

ACKNOWLEDGMENT

This research was supported by the Intramural Research Program of the NIH, National Library of Medicine.

REFERENCES

1. Neveol, A. and Zweigenbaum, P. (2016) Clinical Natural Language Processing in 2015: Leveraging the Variety of Texts of Clinical Interest. Yearb Med Inform. **2016**. 234-239.

2. Kreimeyer, K., Foster, M., Pandey, A., Arya, N., Halford, G., Jones, S. F., et al. (2017) Natural language processing systems for capturing and standardizing unstructured clinical information: A systematic review. J Biomed Inform. **73**, 14-29.

3. Wang, Y., Wang, L., Rastegar-Mojarad, M., Moon, S., Shen, F., Afzal, N. et al. (2018) Clinical information extraction applications: A literature review, J Biomed Inform. **77**, 34-49.

4. Rajkomar, A., Oren, E., Chen, K., Dai., A. M., Hajaj, N.,Hardt, M., et al. (2018) Scalable and accurate deep learning with electronic health records. npj Digital Medicine. **1**. 18.

5. The task overview, https://github.com/ohnlp/fh_eval.

6. Loper, E. and Bird, S. (2002) NLTK: the Natural Language Toolkit. In Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 63-70.

7. Mayo Clinic websites, https://www.mayoclinic.org/[diseases-conditions|symptoms|test-procedures].

8. Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit In Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55-60.

9. Friedman, J. H. (2002). Stochastic gradient boosting. Comput. Stat. Data Anal. **38**, 4 367-378.