TurkuNLP Entry for Interactive Bio-ID Assignment

Suwisa Kaewphan[1,2,3], Farrokh Mehryary[2,3], Kai Hakala[2,3], Tapio Salakoski[1,2] and Filip Ginter[2]
1. Turku Centre for Computer Science, Turku, Finland
2. Department of Future Technologies, University of Turku, Turku, Finland
3. University of Turku Graduate School, University of Turku, Turku, Finland

***Abstract*—We participate in BioCreative VI: Interactive Bio-ID Assignment (Bio-ID) track by developing systems capable of named entity recognition and normalization of 6 entity types, namely *Protein, Cell, Organism, Tissue, Molecule and Cellular*. Our named entity recognition system is based on conditional random fields. For named entity normalization, we apply fuzzy matching and rule-based system to disambiguate and assign unique identifiers to the entities. The official evaluation shows that average F1-scores of all entity types for our recognition and normalization systems on strict span offsets are 0.720 and 0.668, respectively.**

Keywords—CRF; fuzzy matching; rule-based; Simstring

I. Introduction

The main goal of BioCreative VI (Bio-ID) track is to annotate text with the entity types and IDs for organism, gene, protein, miRNA, small molecules, cellular components, cell types and cell lines, tissues and organs, in order to facilitate the curation process. The task principally consists of two major subtasks: i) named entity recognition (NER) and ii) named entity normalization (NEN).

On one hand, several machine learning-based approaches, such as support vector machines and neural networks, have been applied to NER tasks with varying entities ranging from genes to diseases, chemicals and anatomical parts [1,2]. The most recent successful approaches include conditional random field (CRF) classifiers and neural networks [1,3,4]. The approaches for NEN, on the other hand, are largely based on string edit distance and TFIDF weighted vector space representations with a variety of preprocessing approaches to remove the written variations [5,6].

Our system, capable of recognizing all six types of entities and assigning the corresponding identifiers, is based on CRF classifiers, fuzzy matching and a rule-based system. In the following sections we describe our system and its performance based on the official evaluation for both recognition and normalization tasks.

II. Methods

A. Preprocessing

We preprocess the documents by using the publicly available tool [77] converting the character encodings to ASCII. The characters with the missing mapping, such as smiley faces and calendar symbols, are thus replaced with '-' (dash). Subsequently we split the documents into sentences and further tokenize and part-of-speech (POS) tag them using GENIA sentence splitter [8], NERsuite tokenizer and NERsuite POS tagger modules [9], respectively.

Some of the documents contain incorrect word boundaries such as 'mouseliverlysosomes' which should have been written as 'mouse liver lysosome'. While the result of tokenization is overall satisfactory, it is incapable of correctly splitting these words into tokens. We thus resolve this by additional tokenization using the known tokens from the corresponding full-text document. Specifically, we split the tokens using the span of the longest matching document tokens. To reduce the chance of mistakenly tokenizing correct tokens, we only re-tokenize the tokens that belong to noun phrases. Finally, we re-apply POS tagging to complete the data preprocessing.

B. Ontology and controlled vocabularies

We prepare a set of controlled vocabularies and ontologies to assist named entity recognition and normalization. List of concept names and ontologies we used include ChEBI [10] and PubChem [11] (for *Molecule*), Entrez Gene [12] and Uniprot [13] (for *Protein*), NCBI Taxonomy [14] (for *Organism)*, Uberon [15] (for *Tissue*), Cellular Component Ontology [16] (for *Cellular* component) and Cell Ontology (http://purl.obolibrary.org/obo/uberon.owl) and Cellosaurus (http://web.expasy.org/cellosaurus) (for *Cell)*. We preprocess the lists by removing non-alphanumeric characters and lowercasing the symbols.

Specifically for NCBI Taxonomy, we additionally expand the ontology by adding the commonly used abbreviations for scientific names. For binomial nomenclature of names in species rank, we abbreviate the genus while the rest of the names such as species epithet, varieties, strains and substrains, remain the same. For example, 'Escherichia coli O.1197' is abbreviated as 'E. coli O.1197', 'E coli O.1197', 'Es. coli O.1197' and 'Es coli O.1197'. This rule applies to all organisms, except for scientific names of organisms in Viruses and Viroids superkingdoms, since the scientific names do not usually follow binomial nomenclature but are in the form of [Disease] virus [17]. Acronyms are often used as abbreviated scientific names for viruses, for example ZYMV is the acronym of Zucchini yellow mosaic virus, and thus we also add acronyms to the ontology.

C. Named Entity Recognition

For the given training data, we first completely remove annotations for *Assay* entity type and combine *miRNA* and *Gene* with *Protein* annotations. Hence, the total entity counts are 58476, 7476, 6312, 11213, 10604 and 7888 for *Protein*, *Cellular, Tissue*, *Molecule, Cell* and *Organism*. We then randomly partition the training data into a training and a

development set, containing 455 and 115 documents respectively.

We train our NER system on the training set using the NERsuite (http://nersuite.nlplab.org/)—a named entity recognition toolkit—and optimize it against our development set. We train a single CRF model capable of detecting all possible entity types and use micro-averaged F1-score as the optimization metric, derived from the official evaluation script. To achieve higher performance in NER, we directly provide NERsuite with dictionaries through dictionary-tagging module with no further preprocessing or normalization. We compare the performance of different dictionaries on development data using default NERsuite hyperparameters.

For final prediction of the test set, we merge the training and the development sets and re-train the CRF on this data using the best found hyperparameters.

D. Named Entity Normalization and Disambiguation

Our normalization approach is primarily based on fuzzy string matching algorithm where both entity and ontology terms are converted to vectors using character n-gram frequencies. Cosine similarity is then used for calculating similarity between detected entity and ontology terms. In this study, we use Simstring [18], a library for approximate string matching, to retrieve the ontology terms with highest cosine similarity with queried entity, regardless of the type of the terms.

The tagged entities resulting from the NER system are preprocessed using the same approaches we use on dictionaries and ontologies, by removing the punctuations and lowercasing, as described previously. We utilizes approximate string matching approach to all entity types except for *Protein*, which we instead apply 'exact string matching' to retrieve matching identifiers.

Some of the ontology terms are not uniquely linked to a single identifier, but multiple ones. For *Cell*, *Cellular*, *Molecule* and *Tissue*, a random identifier is selected. The selected random identifier is subsequently applied throughout the document. For *Organism* and *Protein*, we develop two separate rule-based systems to uniquely assign an identifier.

For *Organism*, we use taxonomy tree and the following disambiguation rules to assign a taxon identifier to *Organism*. These rules are sequentially applied if the previous rule results in more than one identifier.

1. Take identifier with highest cosine similarity score and its taxonomic rank is under species, which also includes subspecies, strain, variety and no rank.
2. Take identifier of previous mentioned *Organism* if abbreviations match.
3. Take identifier of previous mentioned *Organism* if acronyms match.
4. Take identifier of previous mentioned *Organism* of the same genus.
5. Take identifier of a model organism of the same genus.
6. Take identifier of the most studied organism in PubMed-Central Open Access section.
7. Take a random identifier.

Protein contain the most ambiguous names as the same protein names can be found in multiple organisms if they have the same function or shared sequence identity [19]. Therefore, the information about the *Organism* is crucial for *Protein* normalization. We therefore employ the results of our *Organism* normalization system and use the taxon identifiers to disambiguate *Protein*. However, multiple taxon identifiers can be recognized in a single document, hence we adapt rule-based system proposed by [1] to generate candidate taxon identifiers for the *Protein*. The list of candidate taxon identifiers are ordered according to the following rules.

1. *Organism* mentioned inside *Protein* text span
2. *Organism* mentioned before *Protein* within the same sentence
3. *Organism* mentioned after *Protein* within the same sentence
4. *Organism* mentioned in the previous caption
5. *Organism* mentioned in the same document

In addition, we perform query expansion to generate candidate *Protein* names to cover potential Uniprot and Entrez Gene symbol variations by using stripping algorithm [20]. The algorithm recursively removes common words, such as protein, gene and RNA, and *Organism* from *Protein* to produce a canonical form which includes minimal symbols that are gene symbols in the Entrez Gene database. For instance, 'p53 protein' will result to 'p53'. Finally, the canonical forms are subsequently lower-cased and punctuation-removed. The list of candidate *Protein* names are then ordered by the string length.

For each taxon identifier, we use 'exact string matching' to retrieve corresponding *Protein* identifier. The search starts with the longest candidate *Protein* name and stops when the identifier is found. In case of multiple identifiers, a random one is selected.

III. Result and Discussion

E. Name Entity Recognition

Incorrect word boundaries can result in multiple types of entity annotations for a given token. For example, 'mouseskinfibroblasts' contains the annotations for *Organism*, *Tissue* and *Cell*. Since we train a single CRF-based model to recognize all types of entities, having one token representing multiple entities would have caused the loss of training examples as NERsuite does not support multilabel classification. As mentioned in Method section, we resolve this issue by re-tokenizing the tokens using known tokens from the provided full-text document. The result for recovering the training examples is significant as tokenization from NERsuite alone yields about 97% of the annotations, while this step increases the number of annotations by additional 2pp, equivalent to more than 2000 annotations. As a result, we recover more than 99% of the original annotations with *Organism* with the highest increase in coverage.

TABLE I. COMPARISON OF ANNOTATION COUNTS BETWEEN TOKENIZATION APPROACHES

re-tokenization	*Prot*	*Cellu*	*Tiss*	*Mole*	*Cel*	*Org*
without	97.178	99.772	95.951	96.107	97.099	93.691
with	99.187	99.866	99.842	99.424	99.559	98.921

a. The comparison of annotation counts between preprocessing with only NERsuite tokenization module (without) and with both NERsuite tokenization and additional tokenization (with). The numbers are percents of annotations compared to the provided data presented for each entity type.

It has been demonstrated that domain knowledge, such as controlled vocabularies, is important to attain good performing NER model [3,4]. In this study, we use dictionaries to add features for classifier and compare the model performance on the development data. As shown in Table II, there is no clear performance improvement when adding dictionary features in either strict or overlap modes of evaluation. In the case of cellular component from GO, the performance of NER is however, lower than other models by more than 6pp in F-measure. As a result, we train our model without using any additional dictionary features.

TABLE II. OFFICIAL EVALUATION OF NER SYSTEM ON DEVELOPMENT DATA

Dictionary/ Ontology	**Precision / Recall / F-measure**	
	Strict	***Overlap***
Uberon	**0.787 / 0.688 / 0.734**	**0.882 / 0.771 / 0.823**
ChEBI	0.763 / 0.689 / 0.724	0.865 / 0.780 / 0.821
GO	0.652 / 0.687 / 0.669	0.789 / 0.830 / 0.809
Cellosaurus	0.780 / 0.687 / 0.730	0.875 / 0.772 / 0.820
NCBI Taxonomy	**0.785 / 0.689 / 0.734**	**0.880 / 0.772 / 0.823**
NCBI Gene	0.770 / 0.688 / 0.727	0.870 / 0.778 / 0.821
Cell ontology	**0.788 / 0.687 / 0.734**	**0.883 / 0.770 / 0.823**
No dictionary	0.788 / 0.686 / 0.734	0.882 / 0.769 / 0.822

We finally train NERsuite model on combined training and development sets. The resulting model is subsequently used for tagging the entities in the test dataset. The official evaluation results, shown in Table III, demonstrate that our NER system performs best on *Organism*, achieving F-measure of 0.834. The performance of the system is moderate for *Cell* and *Protein* with F-measure of 0.743 and 0.734, respectively. For the other three entity types, *Tissue*, *Cellular* and *Molecule*, our system shows comparatively lower performances with F-measure of 0.668, 0.642 and 0.579, respectively. *Cellular* proves to be the most difficult entity to recognize. Overall, the performance of model is moderate across all entity types, achieving F-measure of 0.720 and 0.790 on strict and overlap evaluation criteria.

TABLE III. OFFICIAL EVALUATION OF NER SYSTEM ON TEST DATA

Entity	**Precision / Recall / F-measure**	
	Strict	***Overlap***
Cell	0.783 / 0.708 / 0.743	0.841 / 0.760 / 0.799
Cellular	**0.673 / 0.508 / 0.579**	**0.728 / 0.550 / 0.627**
Protein	0.729 / 0.739 / 0.734	0.825 / 0.836 / 0.831
Organism	**0.860 / 0.809 / 0.834**	**0.878 / 0.826 / 0.852**
Molecule	0.775 / 0.587 / 0.668	0.796 / 0.603 / 0.686
Tissue	0.727 / 0.575 / 0.642	0.793 / 0.627 / 0.701
All	**0.747 / 0.694 / 0.720**	**0.821 / 0.762 / 0.790**

F. Name Entity Normalization and Disambiguation

The performance of normalization system is heavily depending on the NER system performance since unrecognized and incorrect spans entities are automatically classified as false negative and false positives, respectively. We thus evaluate our normalization system on the development set based on the gold standard entity mentions to compare the different approaches on different entity types.

TABLE IV. OFFICIAL EVALUATION OF NEN SYSTEM ON DEVELOPMENT DATA

Entity	**Precision / Recall / F-measure**	
	Strict	***Overlap***
Cell	0.902 / 0.946 / 0.923	0.935 / 0.980 / 0.957
Cellular	**0.974 / 0.929 / 0.951**	**0.980 / 0.934 / 0.957**
Protein	0.878 / 0.591 / 0.706	0.902 / 0.606 / 0.725
Organism	0.977 / 0.887 / 0.930	0.993 / 0.901 / 0.945
Molecule	**0.963 / 0.488 / 0.647**	**0.969 / 0.491 / 0.651**
Tissue	0.920 / 0.978 / 0.948	0.930 / 0.988 / 0.958
All	0.914 / 0.700 / 0.793	0.933 / 0.716 / 0.810

Our normalization system performs relatively well on *Cell*, *Cellular*, *Organism* and *Tissue*, where the F-measure ranges from 0.923 to 0.951 under strict criteria. However, the performance of the system drops dramatically on *Molecule* and *Protein*, as their recall of both entities are significantly lower than their precision counterpart. For *Protein*, the exact string matching and a set of taxon identifiers are probably attributing factors for a low recall as these two criteria are probably too stringent resulting in almost half of the *Protein* not being linked to an associated identifier. For *Molecule*, the lower recall is most likely caused by some other factor since the approximate pattern matching was used.

When evaluated against test set, the normalization results differ from gold standard development data as the overall performance is largely depending on the NER system output. As shown in Table V, the normalization performance does not appear to drop drastically even when applied on predicted entities instead of the gold standard mentions.

TABLE V. OFFICIAL EVALUATION OF NEN SYSTEM ON TEST DATA

Entity	Precision / Recall / F-measure	
	Strict	***Overlap***
Cell	0.774 / 0.699 / 0.735	0.830 / 0.750 / 0.788
Cellular	**0.685 / 0.482 / 0.566**	**0.737 / 0.519 / 0.609**
Protein	0.795 / 0.543 / 0.645	0.823 / 0.561 / 0.667
Organism	**0.857 / 0.797 / 0.826**	**0.877 / 0.815 / 0.845**
Molecule	0.787 / 0.581 / 0.668	0.803 / 0.593 / 0.682
Tissue	0.604 / 0.542 / 0.572	0.669 / 0.600 / 0.632
All	0.775 / 0.586 / 0.668	0.809 / 0.612 / 0.697

IV. CONCLUSIONS AND FUTURE WORK

We approach BioCreative Bio-ID task by training a single CRF-based model to recognize all entity types and we link them to their corresponding database identifiers using approximate pattern matching algorithm. For *Protein* and *Organism*, we utilize the ontology structure and surrounding context to disambiguate the entities with multiple identifier candidates. Our systems, evaluated independently, demonstrate a moderate performance overall. However, a lower performance for most types of entities is observed when recognition and normalization are evaluated jointly as the F-score is largely determined by the F-score of the recognition system.

CRF-based classifiers have been a relatively successful tool for entity recognition in biomedical domain, demonstrating state-of-the-art for several entity types. However, it has been recently shown that neural networks with only word embeddings as features can outperform traditional CRF-based NER systems with manually crafted features [1]. Thus, our future work includes developing a neural network-based NER system capable of recognizing multiple types of entities.

Our normalization system for *Protein* and *Molecule* demonstrate a lagging performance when compared with other entities. For *Protein*, applying relaxed string matching in addition to improving the organism assignment algorithm can potentially improve the performance . For *Molecule*, our future work lies on identifying contributing factors that lower the recall and adjusting the system accordingly.

Our current normalization system is limited and time-consuming as it applies several manually generated rules which do not generalize to normalizing other entity types. Thus developing a machine learning system that can be trained on the annotations of new entity type would be an ideal solution for the normalization task. Since the conventions of naming biomedical entities vary among entity types, a unified normalization system can be a challenging task.

ACKNOWLEDGMENT

Computational resources are provided by CSC-IT Center For Science Ltd., Espoo, Finland. This work is supported by ATT Tieto käyttöön grant.

REFERENCES

1. Ding, R., Arighi, C.N., Lee, J.Y., Wu, C.H. and Vijay-Shanker, K. (2015) pGenN, a gene normalization tool for plant genes and proteins in scientific literature. PloS one, 10(8), p.e0135305.

2. Habibi, M., Weber, L., Neves, M., Wiegandt, D.L. and Leser, U. (2017) Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics*, 33(14), pp.i37-i48.

3. Kaewphan, S., Van Landeghem, S., Ohta, T., Van de Peer, Y., Ginter, F. and Pyysalo, S. (2015) Cell line name recognition in support of the identification of synthetic lethality in cancer from text. *Bioinformatics*, 32(2), pp.276-282.

4. Pyysalo, S. and Ananiadou, S. (2014) Anatomical entity mention recognition at literature scale. *Bioinformatics*, 30(6), pp.868-875.

5. Mehryary, F., Hakala, K., Kaewphan, S., Björne, J., Salakoski, T. and Ginter, F. (2017) End-to-End System for Bacteria Habitat Extraction. In *Proceedings of BioNLP 2017*, pp.80-90.

6. Wei, C.H., Kao, H.Y. and Lu, Z. (2015) GNormPlus: An Integrative Approach for Tagging Genes. Gene Families, and Protein Domains, *BioMed Research International*, 918710.

7. Pyysalo, S., Ginter, F., Moen, H., Salakoski, T. and Ananiadou, S. (2013) Distributional Semantics Resources for Biomedical Text Processing. In *Proceedings of LBM 2013.* pp. 39-44.

8. Sætre, R., Yoshida, K., Yakushiji, A., Miyao, Y., Matsubayashi, Y. and Ohta, T. (2007) AKANE system: protein-protein interaction pairs in BioCreAtIvE2 challenge, PPI-IPS subtask. *In Proceedings of the second biocreative challenge workshop*, Vol. 209, p. 212.

9. Tsuruoka, Y., Tateishi, Y., Kim, J.D., Ohta, T., McNaught, J., Ananiadou, S. and Tsujii, J.I. (2005) Developing a robust part-of-speech tagger for biomedical text. In *Panhellenic Conference on Informatics*, pp. 382-392, Springer, Berlin, Heidelberg.

10. Degtyarenko, K., De Matos, P., Ennis, M., Hastings, J., Zbinden, M., McNaught, A., Alcántara, R., Darsow, M., Guedj, M. and Ashburner, M. (2007) ChEBI: a database and ontology for chemical entities of biological interest. *Nucleic acids research*, 36(suppl_1), pp.D344-D350.

11. Bolton, E.E., Wang, Y., Thiessen, P.A. and Bryant, S.H. (2008) PubChem: integrated platform of small molecules and biological activities. *Annual reports in computational chemistry*, 4, pp.217-241.

12. Maglott, D., Ostell, J., Pruitt, K.D. and Tatusova, T. (2010) Entrez Gene: gene-centered information at NCBI. *Nucleic acids research*, 39(suppl_1), pp.D52-D57.

13. UniProt Consortium. (2014) UniProt: a hub for protein information. *Nucleic acids research*, p.gku989.

14. Federhen, S. (2011) The NCBI taxonomy database. *Nucleic acids research*, 40(D1), pp.D136-D143.

15. Mungall, C.J., Torniai, C., Gkoutos, G.V., Lewis, S.E. and Haendel, M.A. (2012) Uberon, an integrative multi-species anatomy ontology. *Genome biology*, 13(1), p.R5.

16. Gene Ontology Consortium. (2004) The Gene Ontology (GO) database and informatics resource. *Nucleic acids research*, 32(suppl 1), pp.D258-D261.

17.. Fauquet, C.M. and Pringle, C.R. (1999) Abbreviations for invertebrate virus species names. *Archives of virology*, 144(11), pp.2265-2271

18. Okazaki, N. and Tsujii, J.I. (2010) August. Simple and efficient algorithm for approximate dictionary matching. In *Proceedings of the 23rd International Conference on Computational Linguistics*. pp.851-859. Association for Computational Linguistics.

19. Chen, L., Liu, H. and Friedman, C. (2004) Gene name ambiguity of eukaryotic nomenclatures. *Bioinformatics*, 21(2), pp.248-256.

20. Van Landeghem, S., Ginter, F., Van de Peer, Y. and Salakoski, T. (2011) EVEX: a PubMed-scale resource for homology-based generalization of text mining predictions. *In Proceedings of BioNLP 2011 workshop*, pp.28-37. Association for Computational Linguistics.