Rule-based Enhancement of Stanza NER

Robert E. Mercer, Mohammed Alliheedi
Department of Computer Science, The University of Western Ontario, London, Canada
Department of Computer Science, Al Baha University, Al Aqiq, Saudi Arabia

***Abstract*—The method adopted uses an already trained model modified with a rule-based system to enhance its performance. Stanza output is modified by rules and the resulting tagged information is mapped back to the original file to find the locations of the chemical names. A mapping of the chemical names to MeSH terms is also done as part of the annotation.**

Keywords—Stanza, rule-based enhancement

I. INTRODUCTION

For the BioCreative VII challenge (Track 2), our team decided to take an already trained model and modify its performance with a rule-based system rather than to design and train a neural model from scratch. This method achieved moderate success. Details are given below.

II. METHODOLOGY

Our proposed method uses Stanza (1,2) followed by a set of post hoc error corrections and then the tagged chemical names in the modified Stanza output are located in the original text file. It is a three step process. First Stanza is used to provide a tokenized text file for each article. There is a set of error corrections that are done to this tokenized file. Details are given below. This tokenized file is then given to Stanza to find the chemical names. There is a set of error corrections that are done to these chemical names. Because the tokenization corrections and chemical name corrections may cause a mismatch between the location in the modified Stanza output file and the location in the original file, the chemical names need to be located in the original file. We now look in detail at these two steps.

A. Tokenization

When investigating the performance of Stanza to find chemical names, it was noticed that many errors are caused by tokenization errors. Chemical names often have multiple parentheses (round and square) and hyphens as part of the name. These symbols are also used as punctuation in the text. And sometimes this punctuation is not separated from the chemical name, so the tokenization step does not always correctly separate these symbols when they should be.

So, the first step was to use Stanza to tokenize the text and then some rules are invoked to modify this tokenization. The initial tokenization step uses the genia package (2). It is used since it does not split chemical names that contain hyphens. The output from this step is then analyzed with the following rules:

1. Separating “,” and “;” at the end of a token when not properly separated (some chemical names include these symbols but only when internal to the name).
2. Separating round and square parentheses, either at the beginning or end of a token, if they are not properly paired.

The first rule is straightforward. The second rule worked reasonably well. However, it failed in a couple of contexts. First, because of the abundance of parentheses, a missing parenthesis is sometimes overlooked in editing of the published text. These typos create a pairing mismatch, so this rule can separate a parenthesis that should be part of chemical name. Second, multi-word parenthetical remarks in the text are dealt with appropriately by this rule, but single-word chemical names as parenthetical remarks have paired parentheses. And multi-word chemical names that contain parentheses can confuse our implementation of this rule. Time did not permit a better implementation of this rule.

B. Chemical name finding

Once the text is tokenized, it is given as input to Stanza with the named entity recognizer processor bc4chemd. Stanza tags each token as “Other”, or as single token chemical names, or as multi-word chemical names using “Begin”, “Internal”, and “End” tags. The output from Stanza is analyzed using the following rules:

1. Stanza sometimes misses chemical names in the text that it has found elsewhere in the text. This rule tags these missed chemical names.
2. Acronyms are not tagged as chemical names by Stanza. This rule finds and tags acronyms as chemical names.
3. Tokens that have a chemical name hyphenated or prefaced with certain modifiers, such as “-based”, “hydroxylated”, are not tagged as chemical names by Stanza. This rule tags such tokens as chemical names. The rule has 22 such modifiers which were manually extracted from the training set. This list can easily be expanded.
4. Chemical names are sometimes followed by words, such as “polymer”, “reductase”. These words are combined with the chemical names. The rule has 15 words which were manually extracted from the training set. This list can easily be expanded.
5. Chemical names are sometimes followed by other items, such as “1d”, “25”. The annotation of these items was inconsistent in the training set. So these items are tagged so that the final chemical name is generated with and without these extra items.
6. Chemical names sometimes contain actions, such as “synthesizing”, “buffered”, which connect two chemical names. These phrases are annotated as a single chemical name. This rule makes this connection and generates a single chemical name.
7. The tokenization step creates tokens (punctuation symbols) that Stanza sometimes tags incorrectly. This rule corrects these incorrect tags.

8. A rule looks for complex chemical name phrases containing the word “and”. The complex phrase ends with words such as “derivatives” or “receptors”, either as the word or the second word following “and”. The rule has 17 words which were manually extracted from the training set. This list can easily be expanded.
9. A rule looks for complex chemical name phrases containing the word “or”. The complex phrase ends with words such as “amine” or “radical”, either as the word or the second word following “or”. The rule has 4 words which were manually extracted from the training set. This list can easily be expanded.
10. Another rule does more processing of parentheses.
11. Stanza does not tag “polymer” or any of its derivatives, so this rule tags these tokens as chemicals.
12. Finally, a rule removes some tokens, such as “(A)”, “(0.5g)”, which have been tagged as chemical names.

Other rules were considered, but they produced too many false positives. Time did not permit making these rules more precise.

C. *Locating and normalizing the chemical name*

The final step is to generate a file containing annotations of the chemical names found. These annotations contain the chemical name text, the offset and length of this text, and the normalized MeSH term for the chemical name for each chemical found in the previous step.

We first describe the normalization process, since it is the most straightforward. We have downloaded the asciimesh c2021.bin and d2021.bin files from which we extract all of the text items that can be associated with each MeSH term. These are loaded in our Python program as a dictionary with which we can easily provide the normalized MeSH term for any chemical name that we have. Because our tagged chemical names can be complex phrases or chemical symbols, these are not found in our dictionary. Chemical name texts not found in our dictionary are given a MeSH term “?”. Time did not permit us to develop a more sophisticated method to deal with these situations.

The last task is to provide the location and length of each chemical name. Two issues caused some problems for us. First, our rules to correct the tokenization adds extra characters to the text (e.g., when separating a “(” used as a punctuation symbol from a following “(” that is part of a chemical name, a space is inserted in the text). Second, the tokenization process reduces the characters in the original text. UTF-8 characters which take two or more bytes in the original text are converted to a single byte character in the tokenized text. As well, occasionally, extra spaces exist in the original text. These are removed in the tokenization. As a consequence, both the location and length in our previous step’s output does not always match these attributes in the original text. When designing our tokenization method, we were unaware that these attributes would be required in the submission file. So, although there are obvious ways to map between the original text and our tokenized text, time constraints did not permit us to design and debug this mapping. Instead we developed an interim (but crude) solution: attempt to realign the original text and the tagged file. This method often works, but when a misalignment cannot be resolved, all of the following chemical names are not locatable in the original file so the location and length attributes are corrupted for all of them.

IV. Results and Discussion

The official results for our competition submission (3) together with the median and benchmark numbers are provided in Fig. 1. The precision numbers for the chemical mention recognition task are reasonably good. The recall values are quite low. Interestingly, our method could find 71.6% of the unique chemical names in the training set, so the recall results being much lower than this is probably due to the test set (4) being biased toward those names that we are unable to find. The precision values for the normalization task are quite good so our method makes reasonably few mistakes. The recall values are low because of the aforementioned low recognition recall and are lower than that task because our method could not find a MeSH ID for a number of chemical names. Another reason that the recall values are low is due to some unforeseen implementation difficulties that were not fixed due to time constraints. A total of 6,616 chemical names are not found in the test text because Stanza converts UTF-8 characters, so a simple equality check will not find the original name. A total of 30,023 chemical names are not found because of misalignment of chemical names in the original text and the Stanza output. If all of these errors were corrected, the recall numbers would increase by 0.0757. We now move to a more in-depth but more speculative discussion of the results.

First and foremost, a good base model to add the rules to must be chosen. Possibly, Stanza (with the genia and bc4chemd packages) is not the best candidate for the competition test set. Its training set may not have been broad enough or the annotation rules used for its training set may have been different. Stanza could find only 61% of the unique chemical names in the competition training set. The rules that we provided increased the recognition level by almost 10 percentage points. However, there may be other reasons that the rules could not increase the recall.

The rules that we provided look only for tokenization errors and chemical name errors that were caused by not combining tokens. Further investigating the errors (after the competition submission) found that some of the errors are caused by not separating chemical names that Stanza was able to recognize. As examples, Stanza annotated the following as chemicals:

1. CHAPSO {3-[(3-cholamidopropyl)dimethylammonio]-2-hydroxy-1-propanesul-fonic acid}
2. ABTS diammonium salt [2,2′-Azino-bis (3-ethylbenzo thiazoline-6-sulfonic acid diammonium salt)]
3. tpy=2,2′;6′,2″−terpyridine

In the first and second items, “CHAPSO” and “ABTS diammonium salt” should have been separated from the following chemical names which should have had the {…} and […] removed since they are being used as parentheses around parenthetical remarks. In the third item tpy is a common abbreviation of 2,2′;6′,2″−terpyridine. The = is just acknowledging this.

Errors of another type were discovered. Some possibly 'more complete' chemical names are found by Stanza in the training set, but the annotation only considers part of the name as a proper chemical. For instance, "oligoglycines" are annotated. Stanza finds a number of names, such as "tetraantennary oligoglycines" which seem (to a non-specialist) to be a more accurate name since the structure of the chemical is included. Another example is "thiophene containing polymers". The word "thiophene" is annotated as a chemical name. Elsewhere in the training set "polymers" is annotated as a chemical name and "containing", as part of annotated chemical names. It is not obvious how to deal with examples such as these.

Finally, some annotation inconsistencies were noticed. An example is a chemical name is sometimes followed by a figure number (to refer to a detailed chemical structure) and sometimes not. Here, we simply labelled both names.

V. Conclusions

In conclusion, our proposed method for the BioCreative VII challenge uses Stanza followed by a set of post hoc error corrections and then the tagged chemical names in the modified Stanza output are located in the original text file. This method has improved somewhat on what Stanza can do to tag the training set provided for the BioCreative VII challenge. The original Stanza is able to find 3520 of the 5595 unique chemical names in the training set for a 62.9% rate. Stanza with the post hoc rule-based corrections is able to find 4005 of the 5595 unique chemical names in the training set for a 71.6% rate.

Acknowledgment

Some of the work reported here was supported by The Natural Sciences and Engineering Research Council of Canada through a Discovery Grant to Robert E. Mercer.

References

1. Qi,P., Zhang,Y., Zhang,Y., Bolton,J. and Manning,C.D. (2020) Stanza: A Python Natural Language Processing Toolkit for Many Human Languages, Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 101–108.
2. Zhang,Y., Zhang,Y., Qi,P., Manning,C.D. and Langlotz,C.P. (2021) Biomedical and clinical English model packages for the Stanza Python NLP library, *Journal of the American Medical Informatics Association*, **28**, 1892–1899.
3. Leaman,R., Islamaj,R., and Lu,Z. (2021). Overview of the NLM-Chem BioCreative VII track: Full-text Chemical Identification and Indexing in PubMed articles, Proceedings of the seventh BioCreative challenge evaluation workshop.
4. Islamaj,R., Leaman,R., Cissel,D., Cheng,M., Coss,C., Denicola,J., Fisher,C., Guzman,R., Kochar,P., Miliaras,N., Punske,Z., Sekiya,K., Trinh,D., Whitman,D., Schmidt,S., and Lu,Z. (2021). The chemical corpus of the NLM-Chem BioCreative VII track: Full-text Chemical Identification and Indexing in PubMed articles, Proceedings of the seventh BioCreative challenge evaluation workshop.

Chemical Mention Recognition

File	Strict			Approximate		
	Precision	Recall	F-score	Precision	Recall	F-score
Track2-Team-114-Subtask1-Run-1.json	0.7219	0.5897	0.6492	0.8348	0.6919	0.7567
Median	0.8476	0.8136	0.8373	0.9220	0.8682	0.8951
Benchmark	0.8440	0.7877	0.8149	0.9156	0.8492	0.8811

Chemical Normalization to MeSH IDs

File	Strict			Approximate		
	Precision	Recall	F-score	Precision	Recall	F-score
Track2-Team-114-Subtask1-Run-1.json	0.8334	0.4645	0.5965	0.8273	0.5279	0.6368
Median	0.7120	0.7760	0.7749	0.6782	0.8402	0.7551
Benchmark	0.8151	0.7644	0.7889	0.7917	0.7889	0.7857

Figure 1: Official results from the BioCreative VII challenge Track 2