HUKB at ChEMU 2021 Task 2: Anaphora Resolution

Kojiro Machi[1], Masaharu Yoshioka[1,2,3]

[1]*Graduate School of Information Science and Technology, Hokkaido University, N14 W9, Kita-ku, Sapporo-shi, Hokkaido, Japan*

[2]*Faculty of Information Science and Technology, Hokkaido University*

[3]*Institute for Chemical Reaction Design and Discovery (WPI-ICReDD), Hokkaido University*

Abstract

This paper describes our system for the ChEMU 2021 task 2 of anaphora resolution for extracting chemical reactions from patents. We divide the task into two subtasks: (1) span detection of mentions that include antecedent and anaphor; and (2) mention classification and relation detection. For the first task, we use a deep learning-based approach. For the second task, we use a rule-based approach that uses features related to chemical reactions by ChemicalTagger, which is a state-of-the-art text-mining tool for chemistry. Our system obtained an exact-match F-score of 0.6907 and a relaxed match F-score of 0.7459.

Keywords

Anaphora resolution, Chemical patents, ChemicalTagger

1. Introduction

Chemical patents are useful information to extract new chemical discoveries because the chemical reactions are usually disclosed in patents [1]. There are several research efforts to extract chemical reaction information from these documents [2, 3].

The ChEMU (Cheminformatics Elsevier Melbourne University) lab aims to develop information extraction techniques for chemical patents and to provide a ChEMU 2021 anaphora resolution task (ChEMU-Ref task) that extracts five types of anaphoric relationships: COREFERENCE, TRANSFORMED, REACTION_ASSOCIATED, WORK_UP, and CONTAINED [4].

Recently, deep learning-based approaches have shown high performance in text mining [5], and the approach has also been applied to coreference resolution [6]. However, it can be difficult for a system that uses only a deep learning-based approach to solve the ChEMU task because this task requires the whole context of a document to detect mentions and relationships. For example, the target compound of a reaction is often written in headings and at the end of procedures. In addition, even if the same words and actions are used in multiple sentences, the label of mentions can be different depending on the objective.

Another approach for extracting chemical information from the chemical literature is a rule-based one. ChemicalTagger [7] is a state-of-the-art system that can recognize chemical named

entities and procedures for a chemical reaction process using predefined rules. In the ChEMU 2020 task, Lowe et al., [8] used this system as a core component and submitted the second-best team results for the event extraction task. Therefore, we assume the information extracted from ChemicalTagger can be used as a clue to identify the relationship for the ChEMU-Ref task.

Therefore, in this paper, we have developed a pipeline system for the task that uses both deep learning-based and rule-based methods. We divided the task into two subtasks: (1) span detection of mentions that include antecedents and anaphora; and (2) mention classification and relation detection. For the first step, the system solves a named entity recognition (NER) problem using BioBERT [9], one of the state-of-the-art deep learning-based methods for NER. For the second step, the system solves a relation detection problem using rules by regex and features of ChemicalTagger, a semantic natural language processing tool for chemical literature.

2. Related Works

2.1. ChemicalTagger

ChemicalTagger is a text-mining tool for chemical literature. ChemicalTagger annotates terms and phrases related to a chemical synthesis procedure using a chemical entity tagger (OS-CAR4 [10]), a regex tagger, and a part-of-speech tagger; then, phrases are parsed by ANTLR [11].

Figure 1 shows examples of parsed trees generated by ChemicalTagger. Both of the Action-Phrases in the figure are identified by verbs defined in dictionaries such as *heated* (<VB-HEAT>) and *diluted* (<VB-DILUTE>). Moreover, chemicals (<OSCARCM>), parameters (<TempPhrase>, <TimePhrase>), and apparatuses (<NN-APPARATUS>) are annotated. As shown in the right-hand tree, <MOLECULE>can be given a role using grammatical information.

```
<Sentence>
 <ActionPhrase type="Heat">
  <NounPhrase>
   <DT-THE>The</DT-THE>
   <JJ-CHEM>sealed</JJ-CHEM>
   <APPARATUS>
    <NN-APPARATUS>tube</NN-APPARATUS>
   </APPARATUS>
  </NounPhrase>
  <VerbPhrase>
   <VBD>was</VBD>
   <VB-HEAT>heated</VB-HEAT>
   <TempPhrase>
    <TO>to</TO>
    <CD>50</CD>
    <NN-TEMP>°C.</NN-TEMP>
   </TempPhrase>
   <TimePhrase>
    <IN-FOR>for</IN-FOR>
    <CD>5</CD>
    <NN-TIME>days</NN-TIME>
   </TimePhrase>
  </VerbPhrase>
 </ActionPhrase>
 <STOP>.</STOP>
</Sentence>
```

```
<ActionPhrase type="Add">
 <NounPhrase>
  <DT-THE>The</DT-THE>
  <JJ-CHEM>reaction</JJ-CHEM>
  <NN-CHEMENTITY>mixture</NN-CHEMENTITY>
 </NounPhrase>
 <VerbPhrase>
  <VBD>was</VBD>
  <VB-DILUTE>diluted</VB-DILUTE>
  <PrepPhrase>
   <IN-WITH>with</IN-WITH>
   <NounPhrase>
    <MOLECULE role="Solvent">
     <OSCARCM>
      <OSCAR-CM>EtOAc</OSCAR-CM>
     </OSCARCM>
    </MOLECULE>
   </NounPhrase>
  </PrepPhrase>
 </VerbPhrase>
</ActionPhrase>
```

Figure 1: Examples of parsed trees generated by ChemicalTagger

2.2. BioBERT

BioBERT [9] is a domain-specific language model pretrained on biomedical documents. BioBERT shows the state-of-the-art scores on several chemical NER datasets. In the original implementation of fine-tuning for the chemical NER task, given tokens are decomposed into subwords and then the labels at the beginning of the tokens are predicted by BioBERT.

3. Method

We have developed a pipeline system for the task that uses both deep learning-based and rule-based methods. While features obtained from ChemicalTagger are useful, the predefined rules of ChemicalTagger are insufficient for mention detection because patent document descriptions vary. In addition, making syntactic patterns for mention detection is expensive. Therefore, for the first step, we used BioBERT, one of the state-of-the-art NER systems based on deep learning, for the span detection of candidates that include antecedents and anaphora. For the second step, we constructed rules for mention classification and relation detection using features generated by ChemicalTagger and regex patterns. Followings are guidelines for constructing rules:

- **Construction and evaluation of rules**: The rules are developed using a training set and evaluated on a development set.
- **Simple rules**: Due to the wide varieties of expressions for anaphora, it is necessary to make complex rules for these expressions to achieve better accuracy. However, it was difficult to construct large number of rules for increasing recall. Therefore, we constructed simple rules that covers the varieties to increase F-score on the development set.

3.1. Candidate Mention Detection

In this stage, we define candidate mention detection as a NER problem. These candidates are used to identify antecedents and anaphora of the previously mentioned five types of relations. We used BioBERT-Base v1.1[1] for the NER model. For the tokenization, we used parsed entities by ChemicalTagger as tokens.

Hyperparameters are the following: max sequence length=384 (this is enough to cover all of sequences contained in training and development sets), batch size=32, learning rate=1e-5, and the number of epoch=50.

The ChEMU-Ref corpus contains discontinuous mentions and overlapped mentions. A method to take a discontinuous mention into account is an extended BIO format using DB and DI for the discontinuous mention (BIOHD format) [12]. We used a similar approach, as shown in Table 1. Here, B and I represent the beginning and inside of the continuous mention, respectively, and DH and DI represent the head and inside of the discontinuous mention, respectively. Because the overlapped mentions are often detected using patterns and features obtained from ChemicalTagger, they are not covered in this stage, and the longest entity is used for training when multiple entities are overlapped.

---

[1]https://github.com/dmis-lab/biobert

Table 1
Example of tokenization labels

Method	dimethyl	formamide	(	DMF	)	(	50	mL	)
BIOHD [12]	DB	DI	O	B	O	DI	DI	DI	DI
Ours	B-DH	I-DH	O	B	O	B-DI	I-DI	I-DI	I-DI

3.2. Relation Detection

In this stage, the candidates extracted in the previous stage are classified into antecedents and anaphora (five types of mentions) and their relations are detected. The relation is extracted by a rule-based system that uses regex and rules from features obtained from ChemicalTagger.

3.2.1. Section Detection

A chemical patent usually consists of three parts: heading, synthesis, and work-up (Figure 2). For the visualizations of the relations, brat annotation tool [13] is used in this paper. Identification

Figure 2: Example of section separation

of these parts is important for this task because it is useful for the relation detection.

To give an example of the advantage of this identification, the heading part does not contain relations, except for COREFERENCE. Another example is when a word is used in the same action phrase; while the word has independent relations, the relation can be detected by checking which part the word belongs to. Therefore, we separated the heading and synthesis sections by finding the first sentence that contains an action phrase except <Synthesis>and separated the synthesis and work-up by finding an action phrase based on previous research [2] (Table 2).

Table 2

Action phrases at the start of the work-up section [2]

Work-up	Concentrate, Degass, Dry, Extract, Filter, Partition, Precipitate, Purify, Recover, Remove, Wash, Quench
Other	Add, ApparatusAction, Cool, Dissolve, Heat, Stir, Synthesize, Wait, Yield

3.2.2. Workflow of Relation Detection

Figure 3 shows an overview of the procedure for relation detection. Because COREFERENCE have different characteristics from the other relations, for example, it can be found by using a simple regex pattern such as "*antecedent* (*anaphor*)", our system detect simple COREFERENCE relations for the first step and the remaining five relations including COREFERENCE for the second step.

For the first step, our system detects simple COREFERENCE relations that can be found by using regex patterns and specific rules. Regex patterns used in this step are shown in Table 3. Because the anaphor found in a discontinuous mention (fifth regex pattern) requires only the chemical substance and the amount in its antecedent, the phrase in the antecedent is extracted by splitting the original noun phrase using prepositions "in, " "of, " and conjunction "and," then, the anaphor is used as the antecedent for the next anaphor.

Next, to find the COREFERENCE of a target compound of that is usually written again at the end of the work-up with its yield, we assume the last mention candidate in the heading to be the target compound and make the relation to the candidate in the work-up with its yield as the COREFERENCE.

Then, for solvent detection that has a specific string and relation, the system ensures the string of a candidate anaphor contains "solvent" or "volatile." When the anaphor contains a keyword, the system searches antecedents from the solvents found by ChemicalTagger.

For the second step, the system detects the remaining relations using the rules and the features generated by ChemicalTagger. Our approach to relation detection starts with classifying mention candidates detected by BioBERT into antecedent and anaphor by using the <OSCARCM> label added by ChemicalTagger. The system classifies the mentions that have an <OSCARCM> label (chemical entity) for antecedent candidates except for the case where the terms exist in the

Table 3

Regex patterns for COREFERENCE detection between <u>antecedent A</u> and <u>anaphor B</u>; dcA represents discontinuous A

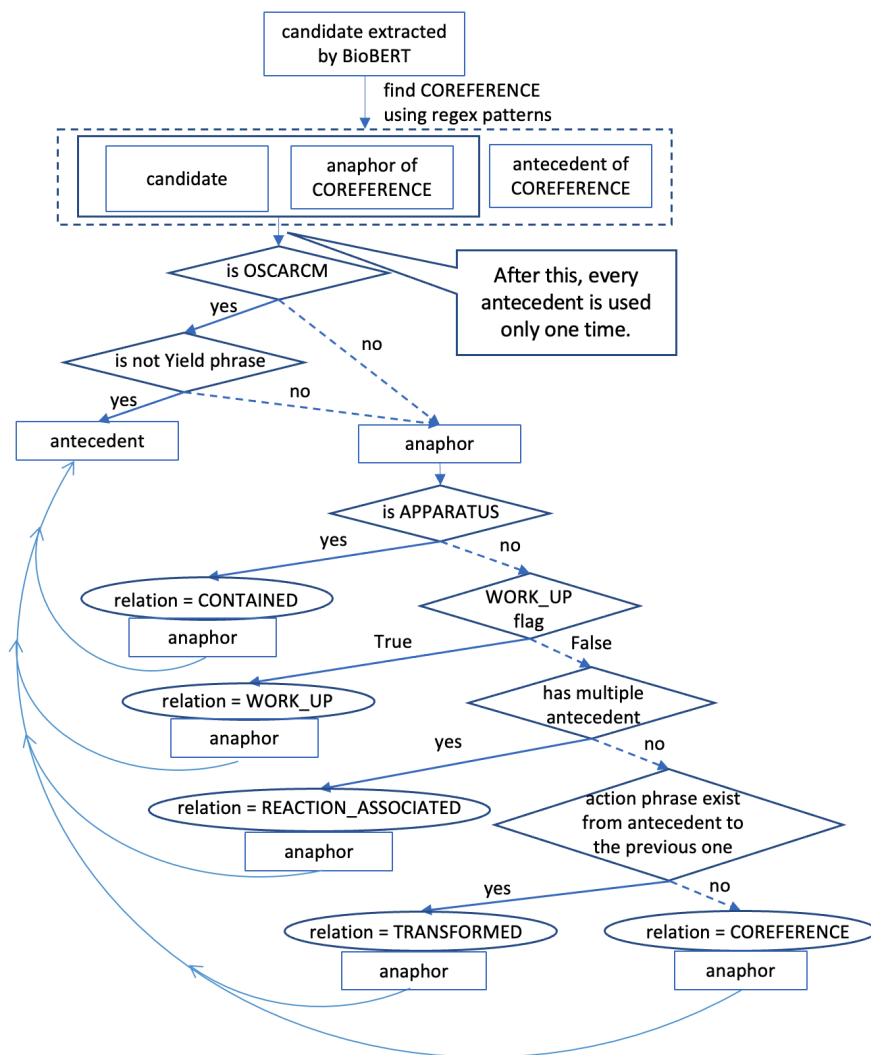| pattern	example
A ([^()]*B[^()]*)	3-Ethynyl-4-hydrocybenzaldehyde (<u>32</u>)
A.{0,20} as B	the title compound (0.166 g, 87%) as <u>a white solid</u>
A: {0,1}B	<u>Intermediate1</u>: <u>5-Bromo-7-chloroindolin-2-one</u>
A\nB\n	<u>Intermediate 11</u>\n<u>2-(tert-bytyl)-5-methoxyisonicotinaldehyde</u>\n
dcA B dcA	dimethyl formamide (<u>DMF</u>) (<u>50mL</u>)

Figure 3: Overview of the procedure for the relation detection

<Yield> phrase (because the chemical entity can be an antecedent in the case that it is written at the end of the work-up as the target compound). Other mentions (usually noun phrases) are classified as anaphor candidates.

After candidates are identified, the system processes them in their order of appearance. When an anaphor is found, the system sets the relationship between the anaphor and antecedent candidates that exist in phrases before the anaphor and that are not connected to the other anaphor. The type of relationship is determined using the procedures described in Figure 3.

For the first step of mention detection, if an anaphor is an <APPARATUS>, the anaphor is assigned a CONTAINED relation because most words in this category contain chemical entities.

Next, the system checks whether the anaphor belongs to the work-up section (when the

WORK_UP flag is True) or not. If it belongs to the work-up section, the anaphor is assigned a WORK_UP relation. The WORK_UP flag is set to True when the action phrase belongs to the work-up phrase (Table 2). This flag is set to False when the Yield phrase with the parameter is found because the phrase usually appears at the end of the work-up section.

Next, the system check counts the antecedent candidates. If there are multiple antecedents, the anaphor is assigned a REACTION_ASSOCIATED relation because the remaining labels, TRANSFORMED and COREFERENCE (except the previously extracted ones by regex and rules from ChemicalTagger), basically have one antecedent.

Finally, the system checks for the existence of any action phrase between an antecedent and the previous phrase. If it exists, the anaphor is assigned a TRANSFORMED relation. In other cases, the anaphor is assigned a COREFERENCE relation because the antecedent is supposed to be unchanged.

3.3. Post-processing

When terms A and B are COREFERENCE and B and C are COREFERENCE, A and C should also be COREFERENCE. To consider this issue, we used a post-processing tool distributed by the organizers[2] that creates a link like that between A and C in the previous example.

4. Main Results

For the evaluation phase, the ChEMU-Ref task employs precision, recall, and F-score calculated by the BRATEval tool[3]. Table 4 shows the evaluation result of our system prediction on the test set. The evaluation was conducted by the task organizers. Here, our system's output was submitted after the evaluation phase.

Our system obtained an exact-match F-score of 0.6907 and a relaxed match F-score of 0.7459.

Table 4
Results of relation detection on the test set

Relation	Exact			Relaxed		
	Precision	Recall	F-score	Precision	Recall	F-score
COREFERENCE	0.6956	0.5319	0.6028	0.7868	0.6016	0.6819
CONTAINED	0.7214	0.6824	0.7014	0.7929	0.7500	0.7708
REACTION_ASSOCIATED	0.6680	0.6803	0.6741	0.7224	0.7357	0.7290
TRANSFORMED	0.6611	0.7169	0.6879	0.6611	0.7169	0.6879
WORK_UP	0.7467	0.7403	0.7435	0.7929	0.7861	0.7895
Overall	0.7132	0.6696	0.6907	0.7702	0.7231	0.7459

5. Error Analysis

In this section, we discuss our system by using a development set. Because our system employs a pipeline approach, we examine candidate mention detection and relation detection.

---

[2]https://raw.githubusercontent.com/yuan-li/chemu2021/master/apply-transitive-closure.py
[3]https://bitbucket.org/nicta_biomed/brateval/src/master/

5.1. Candidate Mention Detection

Table 5 shows the evaluation results of candidate detection by our NER system on the development set in terms of considering overlapped mentions of candidates and when not considering the overlapped mentions.

Our system obtained an exact-match F-score of 0.9885 and a relaxed match F-score of 0.9967 when excluding overlapped entities. Our system obtained an exact-match F-score of 0.9820 and a relaxed match F-score of 0.9902 when considering all entities.

Table 5
Candidate detection results on the development set

Entities	Exact			Relaxed		
	Precision	Recall	F-score	Precision	Recall	F-score
Except overlapped	0.9865	0.9906	0.9885	0.9947	0.9987	0.9967
All	0.9865	0.9776	0.9820	0.9947	0.9857	0.9902

In this stage, the following errors were found:

- Boundary detection by ChemicalTagger tokenization
- Lacking context

There are several cases in which ChemicalTagger failed to identify appropriate terms and sentence boundaries in the tokenization process. "*)(hereinafter*" in document 0735 is an example of term boundary error. In this example, ChemicalTagger extracted this string as one token (annotated as a preposition: <IN>), but the first character *)* should be separated from "*(hereinafter*" to identify the appropriate boundary of a chemical entity.

"*... to give the compound I-1256 (332mg, yield 52%)*" in document 0429 is an example of sentence boundary error. This phrase was split into two sentences because *I-1256* was annotated as a <STOP>token, which is used at the end of a sentence.

There are chemical entities that are not included in relations. Because chemical entities in heading sections are sometimes written as one sentence, our system cannot consider whether candidates in other sentences have relations.

5.2. Relation Detection

Table 4 shows the evaluation results of our system prediction on the development set. Our system obtained an F-score of 0.7569 in relation detection and an F-score of 0.8349 in mention detection for exact match.

Figure 4 shows the confusion matrices of relation detection. Here, the abbreviations are CR (COREFERENCE), CT (CONTAINED), WU (WORK_UP), RA (REACTION_ASSOCIATED) and TR (TRANSFORMED).

In this stage, the following errors were found:

- Boundary detection between the synthesis and work-up sections
- Multiple mentions across relations
- Other issues caused by a lack of rules

Table 6
Results for the development set

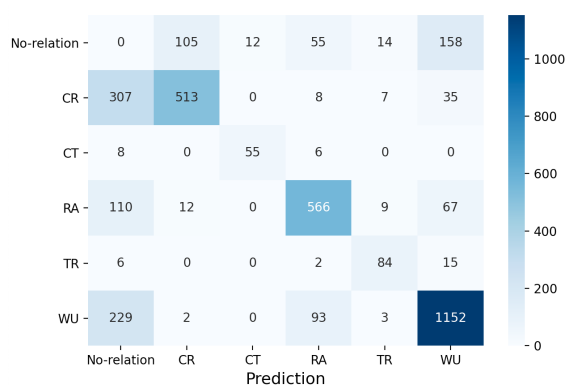| Relation	Mention (anaphor)			Relation		
	Precision	Recall	F-score	Precision	Recall	F-score
COREFERENCE	0.9372	0.7834	0.8535	0.8117	0.5897	0.6831
CONTAINED	0.9118	0.9118	0.9118	0.8209	0.7971	0.8088
REACTION_ASSOCIATED	0.7942	0.7383	0.7652	0.7753	0.7408	0.7577
TRANSFORMED	0.7179	0.7850	0.7500	0.7179	0.7850	0.7500
WORK_UP	0.8241	0.8994	0.8601	0.8073	0.7789	0.7928
All	0.8475	0.8227	0.8349	0.7972	0.7206	0.7569

Figure 4: Relation detection confusion matrices

5.2.1. Boundary Detection between the Synthesis and Work-up Sections on the Development Set

Boundary detection between the synthesis and the work-up section is important for our system because our system classifies an anaphor as WORK_UP when the anaphor is included in the work-up section divided by action phrase (Table 2).

Typical cases of false positives in WORK_UP are caused by the work-up action phrases that are found in the synthesis section. Figure 5 shows an example of this boundary detection error. Because the "Degassed" phrase is used for the detection of the first action phrase of the work-up, our system detects the work-up start from the first sentence. As a result, *the greyish suspension* in the second sentence was annotated as WORK_UP instead of REACTION_ASSOCIATED.

In contrast, false negatives of WORK_UP are boundaries that require knowledge of chemistry. Another type of boundary detection error is the failure to recognize the work-up process shown in Figure 6. In this case, adding chemical entities is common practice for the reaction process, and our system assumes this operation is for the synthesis process. However, it is common to add material that is not directly related to the reaction for the work-up process. It is necessary to have chemical knowledge to determine that this operation is for the work-up process. As a result, *the phases* is annotated as REACTION_ASSOCIATED instead of WORK_UP.
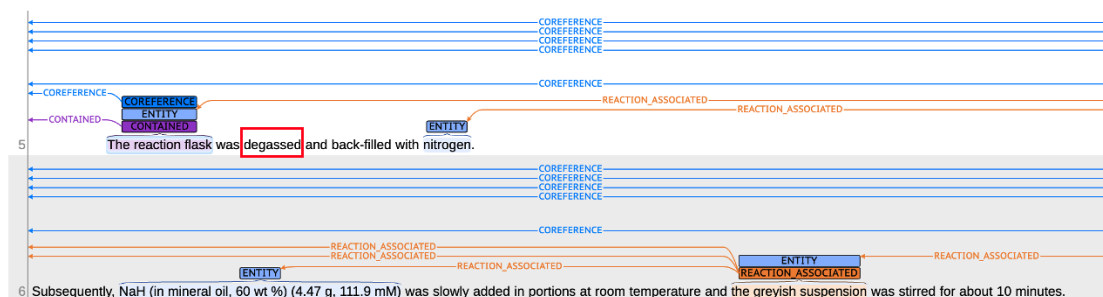
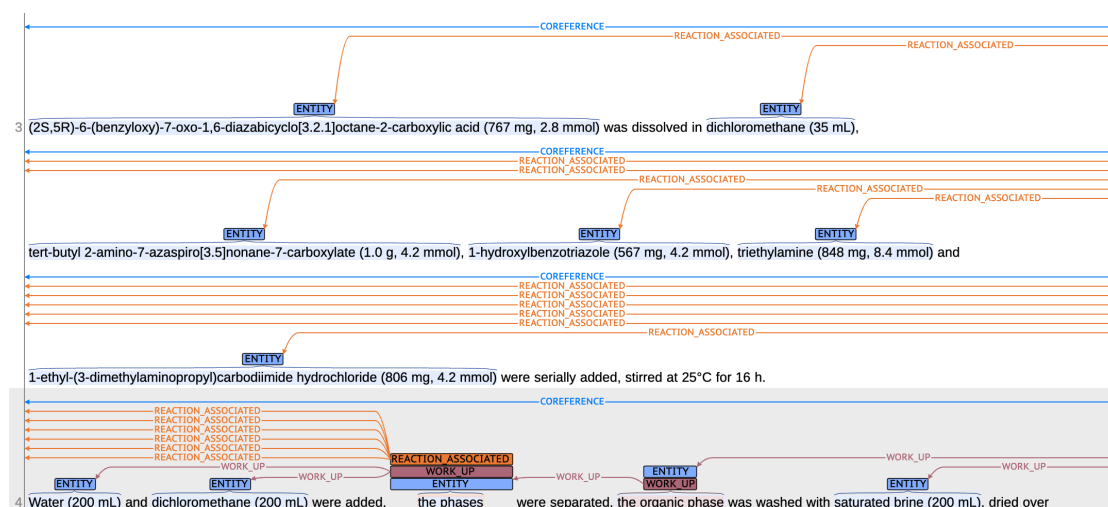**Figure 5:** Work-up phrase in the synthesis section

Figure 6: Boundary of Work-up and overlapped mentions

5.2.2. Multiple Mention Across Relations

Our system cannot detect a mention that has multiple labels except for COREFERENCE because all anaphora is given one label except for COREFERENCE, as shown in Figure 3. For example, *the phases* in Figure 6 is labeled as REACTION_ASSOCIATED and cannot be labeled as WORK_UP.

5.2.3. Other Issues

Other issues we found were as follows.

- As shown in Table 6 and Figure 4, recall of COREFERENCE relations was comparatively lower than others because of an insufficient number of rules. In addition, to improve the recall of the COREFERENCE, we used particular types of phrases that are used for COREFERENCE. We use all solvents (chemical entities) found by ChemicalTagger that

contain the string "solvent" as candidates to identify COREFERENCE. By using this rule, the system generates 47 false positives out of 105 results for no relation.

- Because our relation extraction system relies on the classification of mention candidates (antecedents or anaphora) to set the relations, a mention candidate misclassification causes multiple errors in relation detection. As shown in an example in Figure 7, the misclassification of the term *celite* as an anaphor made two work-up relations to the previous antecedents and removed two work-up relations from the last mention.
- Chemical patents occasionally refer to procedures of other chemical reactions. Our system does not work well in this case because the documents omit detailed procedures.
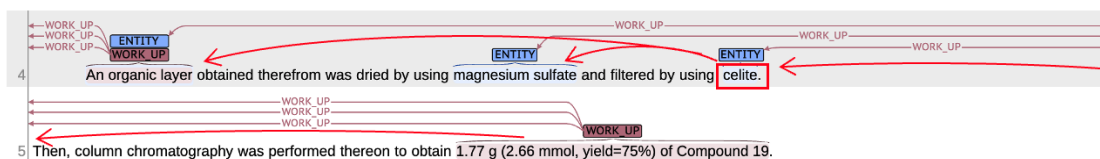

Figure 7: Negative effects of one entity misclassification

6. Conclusion

We propose a hybrid system that uses a state-of-the-art chemical text-mining tool for features and a deep learning-based approach to bridge the gap of entity between the tool and the ChEMU-Ref task.

In the mention candidate detection step, issues related to boundary detection and a lack of context were found. To develop boundary detection, we need to use other tokenization tool that splits a sequence into smaller tokens (e.g. https://github.com/spyysalo/standoff2conll) and construct rules that consider the whole context of a document.

In the mention relation detection step, several issues were found to be caused by gaps of tags set between ChemicalTagger and the ChEMU-Ref task and by insufficient rules in terms of quantity and clarity that we constructed. To develop our system, we need to analyze the gap and improve the rules of the system. For example, it is important to add a tag for mention classification when ChemicalTagger misses the tag such as *celite* in Figure 7. In addition, we need to reconsider the method to detect the work-up section, such as labels for a start of a work-up section and more complex rules. Another direction is using a deep learning-based approach for classifying mentions or end-to-end relation detection itself.

Acknowledgment

References

[1] M. Bregonje, Patents: A unique source for scientific technical information in chemistry related industry?, World Patent Information 27 (2005) 309–315. URL: https://www.sciencedirect.com/science/article/pii/S0172219005000736. doi:https://doi.org/10.1016/j.wpi.2005.05.003.

[2] D. M. Lowe, Extraction of chemical structures and reactions from the literature, Ph.D. thesis, University of Cambridge, 2012.

[3] S. H. M. Mehr, M. Craven, A. I. Leonov, G. Keenan, L. Cronin, A universal system for digitization and automatic execution of the chemical synthesis literature, Science 370 (2020) 101–108. URL: https://science.sciencemag.org/content/370/6512/101. doi:10.1126/science.abc2986. arXiv:https://science.sciencemag.org/content/370/6512/101.full.pdf.

[4] B. Fang, C. Druckenbrodt, S. A. Akhondi, J. He, T. Baldwin, K. Verspoor, ChEMU-ref: A corpus for modeling anaphora resolution in the chemical domain, in: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, Association for Computational Linguistics, Online, 2021, pp. 1362–1375. URL: https://www.aclweb.org/anthology/2021.eacl-main.116.

[5] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[6] K. Lee, L. He, M. Lewis, L. Zettlemoyer, End-to-end neural coreference resolution, arXiv preprint arXiv:1707.07045 (2017).

[7] L. Hawizy, D. M. Jessop, N. Adams, P. Murray-Rust, Chemicaltagger: A tool for semantic text-mining in chemistry, Journal of cheminformatics 3 (2011) 1–13.

[8] D. M. Lowe, J. Mayfield, Extraction of reactions from patents using grammars, in: L. Cappellato, C. Eickhoff, N. Ferro, A. Névéol (Eds.), Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, Thessaloniki, Greece, September 22-25, 2020, volume 2696 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2020. URL: http://ceur-ws.org/Vol-2696/paper_221.pdf.

[9] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, BioBERT: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics (2019). doi:10.1093/bioinformatics/btz682.

[10] D. M. Jessop, S. E. Adams, E. L. Willighagen, L. Hawizy, P. Murray-Rust, Oscar4: a flexible architecture for chemical text-mining, Journal of cheminformatics 3 (2011) 1–12.

[11] T. Parr, The definitive ANTLR reference: building domain-specific languages, Pragmatic Bookshelf, 2007.

[12] B. Tang, Q. Chen, X. Wang, Y. Wu, Y. Zhang, M. Jiang, J. Wang, H. Xu, Recognizing disjoint clinical concepts in clinical text using machine learning-based methods, in: AMIA annual symposium proceedings, volume 2015, American Medical Informatics Association, 2015, p. 1184.

[13] P. Stenetorp, G. Topić, S. Pyysalo, T. Ohta, J.-D. Kim, J. Tsujii, Bionlp shared task 2011: Supporting resources, in: Proceedings of BioNLP Shared Task 2011 Workshop, Association for Computational Linguistics, Portland, Oregon, USA, 2011, pp. 112–120. URL: http://www.aclweb.org/anthology/W11-1816.