The NTNU System at SemEval-2017 Task 10: Extracting Keyphrases and Relations from Scientific Publications Using Multiple Conditional Random Fields

Lung-Hao Lee[1]**, Kuei-Ching Lee**[2]**, Yuen-Hsien Tseng**[1]
[1]Graduate Institute of Library and Information Studies, National Taiwan Normal University
[2]China Development Lab, IBM
[1]No. 162, Sec. 1, Heping East Road, Taipei 10610, Taiwan
[2]No. 13, Sanchong Road, Taipei 11501, Taiwan
lhlee@ntnu.edu.tw, jklee@tw.ibm.com, samtseng@ntnu.edu.tw

Abstract

This study describes the design of the NTNU system for the ScienceIE task at the SemEval 2017 workshop. We use self-defined feature templates and multiple conditional random fields with extracted features to identify keyphrases along with categorized labels and their relations from scientific publications. A total of 16 teams participated in evaluation scenario 1 (subtasks A, B, and C), with only 7 teams competing in all subtasks. Our best micro-averaging F1 across the three subtasks is 0.23, ranking in the middle among all 16 submissions.

1 Introduction

Keyphrases are usually regarded as phrases that capture the main topics mentioned in a given text. Automatically extracting keyphrases and determining their relations from scientific articles has various applications, such as recommending articles to readers, matching reviewers to submissions, facilitating the exploration of huge document collections, and so on. An adapted nominal group chunker and a supervised ranking method based on support vector machines have previously been used to extract keyphrase candidates (Eichler and Neumann, 2010). The conditional random field based keyphrase extraction method has been presented (Bhaskar et al., 2012). A naïve approach has been proposed to investigate characteristics of keyphrases with section information from well-structured scientific articles (Park et al., 2010). Features broadly used for the supervised approaches in scientific articles have been assessed in the compilation of a comprehensive feature list (Kim and Kan, 2009). Maximal sequences and page ranking have been combined to discover latent keyphrases within scientific articles (Ortiz et al., 2010). Noun phrases containing multiple modifiers have been extracted from earth science publications and generalized by matching tree patterns to the syntax trees of the sources texts (Marsi and Öztürk, 2015). Keyphrase boundary classification has been regarded as a multi-task learning problem using deep recurrent neural network (Augenstein and Søgaard, 2017).

The ScienceIE task seeks solutions to automatically identify keyphrases within scientific publications, label them, and determine their relationships. Specifically, the ScienceIE task contains three subtasks: (A) *Identification of keyphrases*: to identify all the keyphrases within a given scientific publication; (B) *Classification of identified keyphrases*: to label each keyphrase as Process, Task, or Material; (C) *Extraction of relationships between two identified keyphrases*: to label keyphrases as Hyponym-of or Synonym-of.

The ScienceIE task presents three evaluation scenarios. In Scenario 1, only plain text is given for subtasks A, B, and C; in Scenario 2, plain text with manually annotated keyphrase boundaries are given for subtasks B and C; and in Scenario 3, plain text with manually annotated keyphrases and their types are given for subtask C. System output is matched against a gold standard to measure system performance. The micro-averaging precision, recall, and F1 across the subtask(s) are used in the task. Each participating team can submit at most

Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017), pages 951–955,
Vancouver, Canada, August 3 - 4, 2017. ©2017 Association for Computational Linguistics

three results and the best result for each evaluation scenario is taken as the performance of the participating team.

This article describes the NTNU (National Taiwan Normal University) system for the ScienceIE task at the SemEval 2017 workshop. Our solution uses multiple conditional random fields at the sentence level. Each sentence is parsed to obtain features, including words, lemmas, part-of-speech tags, and syntactic phrases. CRFs are then trained to learn sequential patterns using the datasets provided by task organizers. We participated in the evaluation scenario 1 with three subtasks. Our best micro-averaging F1 of 0.23 ranked in the middle of all 16 submissions.

The rest of this paper is organized as follows. Section 2 describes the details of the NTNU system for the ScienceIE task. Section 3 presents the evaluation results and performance comparisons. Section 4 discusses some findings. Conclusions are finally drawn in Section 5.

2 The NTNU System

Our proposed approach uses the Conditional Random Field (CRF) technique (Lafferty et al., 2001), a type of discriminative probabilistic graph model, by learning linguistically motivated features to extract the keyphrases from scientific articles and identify their relations. The linear chain CRF is empirically effective for predicting the sequence of labels given a sequence input. A word in a sentence is regarded as a state in our CRF. Given an observation and its adjacent states in terms of the distinguished features, the probability of reaching a state is determined based on the Stochastic Gradient Descent. In the testing phase, the proposed CRF reports the sequence of categories with the largest probability as the identified result.

The following four features are used for training the CRF model with the Stanford CoreNLP toolkit (Manning et al., 2014).

- *Word*: the original words in the sentence of a scientific article are directly used without any revision.
- *Lemma*: this is to reduce inflectional forms and derivationally related forms to determine the lemma of a word in terms of its intended meaning
- *Part-of-Speech*: noun, verb, adjective, adverb, pronoun, etc.

Word	Lemma	POS Tag	Syntactic Phrase
This	this	DT	S-NP
paper	paper	NN	S-NP
addresses	address	VBZ	x
the	the	DT	NP-NP
tasks	task	NNS	NP-NP
of	of	IN	x
named	name	VBN	NP-NP
entity	entity	NN	NP-NP
recognition	recognition	NN	NP-NP
(	-lrb-	-LRB-	x
NER	ner	NN	PRN-NP
)	-rrb-	-RRB-	x
.	.	.	x

Table 1: An example sentence with features.

Token	Task	Pro.	Mat.	Syn.	Hyp.
This	O	O	O	O	O
paper	O	O	O	O	O
addresses	O	O	O	O	O
the	O	O	O	O	O
tasks	O				
of	O	O	O	O	O
named	Task	O	O	Syn.	O
entity	Task	O	O	Syn.	O
recognition	Task	O	O	Syn.	O
(	O	O	O	O	O
NER	Task	O	O	Syn.	O
)	O	O	O	O	O
.	O	O	O	O	O

Table 2: An example sentence with encoding.

- *Syntactic Phrase*: a phrasal category which is a type of syntactic unit in the grammar structure. Noun phrases are usually regarded as keyphrases in scientific texts. Hence, we only adopt noun phrases and their upper phrasal category as features.

Table 1 shows an example sentence with its corresponding features. Each row denotes a token in the sequence. In addition to words, the remaining three features (i.e., lemmas, part-of-speech tags, and syntactically phrasal tags) are provided by the Stanford CoreNLP toolkit.

Table 2 shows the same example sentence with encoding for training multiple CRF models. We use the simplest IO encoding, which tags each token as either being in a particular type of keyphrase X or in no keyphrase (denoted as "O").

We regard the relations Synonym-of and Hyponym-of as individual types in this sequential labeling problem. The one-vs.-rest strategy, which involves training a single classifier per class, is adopted using class samples as positive instances and all the other samples as negatives. In total, we have five corresponding CRF models for each type (i.e., Task, Process, Material, Synonym-of, and Hyponym-of).

During the testing phase, all trained CRF models are parallel to label one of types. The tags predicting by both Synonym-of and Hyponym-of CRF models are reliable dependently on the other three models, because pairs of keyphrase should be identified first for relations. Hence, we check the pairs of keyphrases to keep those are identified by Task, Process and Material CRF models. Finally, we integrate all identified results as our system outputs without handling any conflicts.

3 Evaluation

3.1 Data

The datasets for the ScienceIE task were provided by task organizers (Augenstein et al., 2017). The collected corpus consisted of journal articles from ScienceDirect open access publications evenly distributed among Computer Science, Material Science and Physics. The training, development, and test datasets were comprised of sampled paragraphs, of which 350 were used for training data, 50 for development, and 100 for testing. These datasets were made available to participants without copyright restrictions.

No external resources were used to supplement the datasets. To pre/post-process the datasets, we transformed alphabet-based start/end counts into word-based positions.

3.2 Implementation

The CRF++ toolkit was used for system implementation. CRF++ is an open source implementation of conditional random fields for segmenting or labeling sequential data, and is available at https://taku910.github.io/crfpp/

Supplementary Material in the Appendix shows feature templates used in our implemented system. Each line denotes one template, in which the first characters "U" and "B" respectively represent unigram and bigram features. In each template, a special macro %[row, col] is used to specify a token in the input data, in which row specifies the relative position from the current focus token and col specifies the absolute position of the column.

Type	Precision	Recall	F1
Task	0.17	0.05	0.08
Process	0.44	0.17	0.25
Material	0.47	0.19	0.27
Synonym-of	0.73	0.07	0.13
Hyponym-of	1.00	0.01	0.02

Table 3: Our results for each type.

Type	Precision	Recall	F1
Subtask A only	0.53	0.21	0.30
Subtask A+B only	0.43	0.17	0.24
Subtask C only	0.75	0.04	0.08
Subtask A+B+C	0.44	0.16	0.23

Table 4: Our results for each subtask.

The encoding scheme we used was one-hot. We had 5 columns, where the first four ones respectively denoted features, *i.e.*, Word, Lemma, Part-of-Speech and Syntactic Phrases, and the last was a given type, *e.g.*, Process or not, for training a specific CRF model to label a given type. In the testing phase, the same template file was used and the last column was an estimated type predicting by the trained CRF model.

3.3 Metrics

The traditional metrics precision, recall, and F1-score were computed to measure system performance for each subtask. The micro-averaging strategy was then used to obtain overall score across subtask(s).

3.4 Results

Table 3 shows our results for each defined type. "Task" for subtask B and "Hyponym-of" for subtask C clearly performed worse than other three types.

Table 4 shows our results for each subtask. Comparing subtask C with subtasks A and B shows the former is relative more difficult.

3.5 Comparisons

Of the total 16 submissions, 9 teams did not participate in subtask C. We participated in all subtasks, achieving a micro-average F1 of 0.23, thus ranked 9th of the 16 submissions.

4 Discussion

For this task, we only use multiple CRF models with four defined features. In addition to the Stanford CoreNLP toolkit for extracting features, we do not use any other methods such as the NER tool. Our error analysis reflects that the NER may be useful to improve the performance of Task keyphrase identification. It is also difficult to extract the Hyponym-of relation due to the limitation of long distance using existing features templates.

During the development phrase, we attempted to identify the relations between extracted phrases using manually crafted rules. Our multiple CRF models with the help of rules improved the performance on the development set, but performed worse on the testing set. Hence, we do not adopt rules in the system module. Our observations suggest that human-crafted rules do not perform well due to the challenge of coverage.

5 Conclusions

This study describes the NTNU system in the ScienceIE task, including system design, implementation and evaluation. This is our first exploration of this research topic. Future work will explore other features to further improve performance.

Acknowledgments

This study was partially supported by the Ministry of Science and Technology, under the grant MOST 105-2221-E-003-020-MY2 and the “Aim for the Top University Project” and “Center of Learning Technology for Chinese” of National Taiwan Normal University, sponsored by the Ministry of Education, Taiwan.

References

Isabelle Augenstein, and Anders Søgaard. 2017. Multi-task learning of keyphrase boundary classification. In *Proceedings of ACL 2017, the 55th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Isabelle Augenstein, Mrinal Kanti Das, Sebastian Riedel, Lakshmi Nair Vikraman, and Andrew McCallum. 2017. SemEval 2017 task 10: ScienceIE – Extracting keyphrases and relations from scientific publications. In *Proceedings of SemEval 2017, the 11th International Workshop on Semantic Evaluatio*n. Association for Computational Linguistics.

Pinaki Bhaskar, Kishorjit Nongmeikapam, and Sivaji Bandyopadhyay. 2012. Keyphrase extraction in scientific articles: a supervised approach. In *Proceedings of COLING 2012, the 24th International Conference on Computational Linguistics: Demonstration Papers*. IIT Bombay and Association for Computational Linguistics, pages 17-24. http://aclweb.org/anthology/C12-3003

Kathrin Eichler, and Günter Neumann. 2010. DKFI KeyWE: ranking keyphrases extracted from scientific articles. In *Proceedings of SemEval 2010, the 5th International Workshop on Semantic Evaluation*. Association for Computational Linguistics, pages 150-153. http://aclweb.org/anthology/S10-1031

Su Nam Kim, and Min-Yen Kan. 2009. Re-examining automatic keyphrase extraction approaches in scientific articles. In *Proceedings of MWE 2009, the 5th Workshop on Multiword Expressions: Identification, Interpretation, Disambiguation and Applications*. Association for Computational Linguistics, pages 9-16. http://aclweb.org/anthology/W09-2902

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML 2001, the 18th International Conference on Machine Learning*. Morgan Kaufmann Publishers, pages 282-289. http://dl.acm.org/citation.cfm?id=655813

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of ACL 2014, the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Association for Computational Linguistics, pages 55-60. http://aclweb.org/anthology/P14-5010

Erwin Marsi, and Pinar Öztürk. 2015. Extraction and generalization of variables from scientific publications. In *Proceedings of EMNLP 2015, the 2015 conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 505-511. http://aclweb.org/anthology/D15-1057

Roberto Ortiz, David Pinto, Mireya Tovar, and Héctor Jiménez-Salazar. 2010. BUAP: an unsupervised approach to automatic keyphrase extraction from scientific articles. In *Proceedings of SemEval 2010, the 5th International Workshop on Semantic Evaluation*. Association for Computational Linguistics, pages 174-177. http://aclweb.org/anthology/S10-1037

Jungyeul Park, Jong Gun Lee, and Béatrice Daille. 2010. UNPMC: native approach to extract keyphrases from scientific articles. In *Proceedings* *of SemEval 2010, the 5th International Workshop on Semantic Evaluation*. Association for Computational Linguistics, pages 178-181. http://aclweb.org/anthology/S10-1038

A Supplementary Material

The feature templates used for training CRF models are shown as follows.

```
#Unigram
U01:%x[-2,0]
U02:%x[-1,0]
U03:%x[0,0]
U04:%x[1,0]
U05:%x[2,0]
U06:%x[-2,0]/%x[-1,0]
U07:%x[-1,0]/%x[0,0]
U08:%x[0,0]/%x[1,0]
U09:%x[1,0]/%x[2,0]
U11:%x[-2,1]
U12:%x[-1,1]
U13:%x[0,1]
U14:%x[1,1]
U15:%x[2,1]
U16:%x[-2,1]/%x[-1,1]
U17:%x[-1,1]/%x[0,1]
U18:%x[0,1]/%x[1,1]
U19:%x[1,1]/%x[2,1]
U21:%x[-2,2]
U22:%x[-1,2]
U23:%x[0,2]
U24:%x[1,2]
U25:%x[2,2]
U26:%x[-2,2]/%x[-1,2]
U27:%x[-1,2]/%x[0,2]
U28:%x[0,2]/%x[1,2]
U29:%x[1,2]/%x[2,2]
U31:%x[-2,3]
U32:%x[-1,3]
U33:%x[0,3]
U34:%x[1,3]
U35:%x[2,3]
U36:%x[-2,3]/%x[-1,3]
U37:%x[-1,3]/%x[0,3]
U38:%x[0,3]/%x[1,3]
U39:%x[1,3]/%x[2,3]
#Bigram
B
```