NTNU-1@ScienceIE at SemEval-2017 Task 10: Identifying and Labelling Keyphrases with Conditional Random Fields

Erwin Marsi, Utpal Skidar, Cristina Marco, Biswanath Barik, Rune Saetre
Department of Computer Science
Norwegian University of Science and Technology
{emarsi,utpals,cristina.marco,biswanath.barik,saetre}@ntnu.no

Abstract

We present NTNU's systems for Task A (prediction of keyphrases) and Task B (labelling as Material, Process or Task) at SemEval 2017 Task 10: Extracting Keyphrases and Relations from Scientific Publications (Augenstein et al., 2017). Our approach relies on supervised machine learning using Conditional Random Fields. Our system yields a micro F-score of 0.34 for Tasks A and B combined on the test data. For Task C (relation extraction), we relied on an independently developed system described in (Barik and Marsi, 2017). For the full Scenario 1 (including relations), our approach reaches a micro F-score of 0.33 (5th place). Here we describe our systems, report results and discuss errors.

1 Approach

We choose Conditional Random Fields (Lafferty et al., 2001) because they have produced state-of-art results on comparable sequence labelling tasks such as named entity recognition in biomedicine. Two systems were developed, using different feature sets and alternative CRF implementations.

Preprocessing Input text is linguistically analysed using the Spacy NLP pipeline (Honnibal and Johnson, 2015), including sentence splitting, tokenisation, lemmatisation and dependency parsing. Since CRFs cannot handle Brat's stand-off annotation format directly, keyphrase annotations are first converted to the Inside-Outside-Begin (IOB) tagging scheme by aligning their character offsets with the character offsets of tokens: if the start character offset of a token coincides with the start offset of an annotated keyphrase, the token receives a B (begin) tag; if the offset span of a token falls within the offset bounds of a keyphrase, the token gets an I (inside) tag, otherwise the token is assigned an O (outside) tag. Each sentence corresponds to a sequence of IOB tags, serving as the labelled sequence for the CRF. Separate IOB tags are derived for each of the three keyphrase classes (Material, Process, Task). Annotations and tokens do not always properly align; the resulting errors are discussed in Section 3.

System 1 relies on the CRFsuite implementation (Okazaki, 2007) as wrapped by the sklearn-crfsuite module for SciKit Learn. A dedicated classifier is trained for each of the three keyphrase classes. CRFs are used with default parameter setting. The following features were selected per class by cross-validation on the development data:

- Word features: word shape (e.g. 'Xxxx'), is-alpha, is-lower-case, is-ascii, is-capitalized, is-upper-case, is-punctuation, like-number, prefix-chars (2,3,4), suffix-chars (2,3,4), is-stopword, all in a window of size 3 for Material and Process;
- Lemma and POS, in a window of size 5 for Material and Process, in window of size 3 for Process;
- Wordnet (for Material only): synset names of all hypernyms (transitive closure), in a window of size 5

Supervised learning is generally hampered by skewed class distributions, where minority classes tend to be predicted poorly. In our case, the O tag is by far the most frequent tag. To reduce its weight, all sentences without a Material keyphrase are removed from training material of the CRF for predicting the Material class, and likewise for the other two classes.

Output is postprocessed with the intention of improving consistent labelling throughout a single

Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017), pages 938–941,
Vancouver, Canada, August 3 - 4, 2017. ©2017 Association for Computational Linguistics

text. For example, if a majority of the occurrences of the phrase '*carbon*' in a text is labelled as Material, then any unlabelled occurrences are by extension also labelled as Material.

System 2 consists of two steps: (1) detection of keyphrase boundaries; (2) labelling of keyphrases. Both steps are implemented using the C++ based CRF++ package[1]. The boundary detection model uses the the following features: local context: -2 to +2[2], POS, lemma, prefix-suffix-chars (1,2,3,4), is-word-length-with-upper-case $<$ 5, word-frequency, shape, is-stopword, is-all-upper-case, is-beginning-upper-case, is-inner-upper-case, is-single-upper-case, is-words-in-training-data, is-all-digit and is-alpha-digit.

For labelling of keyphrases, separate classifiers are trained for each class, where the classifiers for Process and Task (but not Material) use the predicted keyphrase boundaries as a feature. The following features were used for Material:

- Word features: local-context (uni-gram and bi-gram, -2 to +3), is-all-digit (-1 to +2), is-single-upper-case (-2 to +2), is-all-upper-case (-2 to +1), is-inner-upper-case (-2 to +4), is-stopword (-1 to +3), shape (-2 to +1), prefix-chars (1), suffix-chars (1,2,3), is-word-length-with-upper-case $<$ 5 (-2 to +3), is-word-in-training-data (-3 to +3)
- Babelfy Mention (-2 to +2): Checks if current word belongs to any Babelfy (Moro et al., 2014) named entities
- Lemma (-1 to +3) and POS (-1 to +2)
- Wordnet: Synonym and Hypernym (first 2 synset names and hypernyms of first and third synset names. If no hypernyms are found, we represent it as ND (not defined)).

The following features were used for Process:

- Word features: local-context (uni-gram and bi-gram: -4 to +2), is-digi-alpha (-1 to +4), is-all-digit (-3 to +3), is-inner-upper-case (-1 to +1), is-beginning-upper-case (-2 to +4), is-all-upper-case (-2 to +2), is-stopword (-2 to +1), shape (-2 to +3), word-frequency (-4 to +4), is-word-in-training-data (-3 to +1), prefix-chars (1,3), suffix-chars (1,3)
- keyphrase boundary according to boundary detection model (-2 to +4)
- POS (uni-gram and bi-gram: -4 to +1)
- Wordnet: Synonym and Hypernym (second synset names and hypernyms of first and third synset names)

The following features were used for Task:

- Word features: local-context (uni-gram and bi-gram, -4 to +1), is-digi-alpha (-3 to +3), is-all-digit (-3 to +4), shape (-1 to +4), is-word-in-training-data (-1 to +4), prefix-chars (1,4), suffix-chars (1,3,4)
- keyphrase boundary according to boundary detection model (-2 to +3)
- Babelfy Mention (-3 to +1)
- Lemma and POS (-4 to +3)
- Wordnet: Synonym and Hypernym (first synset names and hypernyms of fourth synset name)

System 3 System 3 is an optimal combination of the two preceding systems according to CV on the development data. Based on the precision value, System 2 was given higher priority when both systems identified the words as keyphrases. That is, we add any Task or Material keyphrases predicted by System 1 to those predicted by System 2, unless they happen to overlap with any System 2 predictions (Process remained unaltered).

IOB-to-Brat conversion The final step consists of merging the IOB tags predicted by the three separate models in order to produce labelled keyphrases in Brat format.

Experimental setup Cross-validation on the training data was used to select features and tune hyper-parameters. The best performing systems were tested on dev data to check for undesired overfitting. Finally the best systems were trained on the combination of train and dev data to make predictions on test data.

Relation extraction For Task C (relation extraction), we relied on an independently developed system described in (Barik and Marsi, 2017), which performs exhaustive pairwise classifications of keyphrase pairs of the same type within a sentence.

2 Results

Results for our three systems are shown in Table 1. Micro averages are weighted across the

[1] https://taku910.github.io/crfpp/

[2] Here '-' and '+' indicate the number of preceding and following words in the context window respectively.

three labels for keyphrases and the two relation types, but as the keyphrases are substantially more frequent, the weight of the relations is relatively small. System 1 performs worst and system 2 performs best, although the differences are small. System 1 mainly wins on precision. The combination of both in system 3 does not offer any advantages, except for higher recall. All system obtain best scores for Material and worst scores for Task. This can be partly explained by the support for each class: Material and Process instances are much more frequent than Task in the training data. Another part of the explanation may be that Process and Task are harder to distinguish from each other.

Results on test data are substantially lower than on the dev data, with 6 to 7 percent lower average F-scores. This suggests that the models were overfitted on the combination of train and dev data. This is somewhat surprising, because no such differences showed up between cross-validated scores on the training data and scores on the dev data.

Performance on relation extraction is rather poor when compared with the scores obtained with manually annotated keyphrases as input. This is to be expected, as errors in keyphrase extraction propagate to errors in relation extraction. For more analysis of the relation extraction system, see (Barik and Marsi, 2017).

3 Discussion

IOB tags The offsets of annotated phrases did not always properly align with the beginning or end of a token. This was partly due to tokenisation errors. In particular, Spacy tended to consider periods as part of an abbreviation instead of the end of a sentence. For example, it took the period after '*Co(II)OEP*.' as a part of an abbreviation rather than a sentence ending, which does not align with the annotated phrase '*Co(II)OEP*'. Likewise, words compounded with a dash or slash (e.g. '*solid-liquid*') were sometimes individually annotated as keyphrases, but not split by Spacy, or the other way around. There were also errors were annotators did not include all characters in the text span (e.g., '*ossil mass*' instead of '*fossil mass*', or unintentionally included extra characters (e.g. '*EBL and HSQ development, t*').

In order to estimate the impact of IOB conversion errors on the scores, we converted annotated keyphrases in Brat format to IOB format and then back to Brat format. We then used the eval.py script to compute the scores of the resulting 'predictions'. The number of misalignments and their impact on precision, recall and F-score are shown in Table 2. We conclude that the impact of conversion to IOB tags on F-score is relatively small: between 1 to 3 percent at maximum, assuming all predictions are correct.

Failed attempts We tried tuning the CRF hyper-parameters using grid search (for run 1), optimising the micro-average F-score over the B and I tags. However, this criterion did not correspond well with the official scores reported by eval.py. In fact, CRFs with optimised hyper-parameters yielded official scores that were lower than for CRFs with default parameter setting. Optimising directly on the official scores is more expensive and complicated, because of the conversion of IOB tags to Brat annotation. However, doing so may improve performance.

Qualitative error analysis The analysis of errors has been conducted over a random sample of 10% of the documents from the test data under the best system (2). This analysis shows that almost half of the errors are words or phrases incorrectly tagged as keyphrases. The other half are due to either incorrect boundaries (19%), such as *ERP system* instead of *hybrid ERP system* in S0166361516300926; label (18%), e.g. *FIB instruments* as Material instead of Process in S0168583X14003929; or both incorrect boundaries and label (15%), e.g. *finding a group of optimized coefficients* in S0021999113002945 is automatically annotated as Process whereas *optimized coefficients* is Material in the test data.

Other types of errors are those in which the same phrase has been annotated with two different labels and only one of these is correct. For example, *SNR* (S096386951400070X) or *DP* (S0010938X15301268) are both Material and Process, but only the former exists in the gold standard data. This is especially frequent among acronyms.

It is worth mentioning that part of these errors are also due to errors already present on the annotated test data. For instance, *RH ceramics* in *value of the fracture toughness of RH ceramics* is clearly some kind of material, but it is unlabelled in the gold standard data.

Table 1: Results on dev and test data

System		Label	Dev			Test		
			Prec	Rec	F	Prec	Rec	F
System 1		Material	0.48	0.47	0.47	0.36	0.42	0.39
		Process	0.38	0.34	0.36	0.35	0.28	0.21
		Task	0.17	0.12	0.14	0.08	0.09	0.09
		Synonym-of	0.40	0.22	0.29	0.67	0.18	0.28
		Hyponym-of	0.29	0.08	0.13	0.05	0.03	0.04
	Micro Average		0.41	0.34	0.37	0.32	0.31	0.31
System 2		Material	0.54	0.40	0.46	0.41	0.40	**0.41**
		Process	0.44	0.33	0.38	0.39	0.29	**0.33**
		Task	0.18	0.16	0.17	0.10	0.12	**0.11**
		Synonym-of	0.41	0.27	0.32	0.65	0.21	0.32
		Hyponym-of	0.39	0.11	0.17	0.11	0.05	**0.07**
	Micro Average		0.45	0.32	0.37	**0.36**	0.31	**0.33**
System 3		Material	0.45	0.53	0.49	0.34	0.49	0.40
		Process	0.45	0.31	0.37	0.38	0.27	0.32
		Task	0.17	0.21	0.19	0.07	0.13	0.09
		Synonym-of	0.38	0.29	0.33	0.64	0.22	**0.33**
		Hyponym-of	0.26	0.11	0.16	0.05	0.05	0.05
	Micro Average		0.40	0.38	0.39	0.31	**0.34**	0.32

Table 2: IOB alignment errors and their impact

	#spans	#misalign	Prec	Rec	F-score
train	6721	138 (2.1%)	-0.01	-0.03	-0.02
dev	1154	16 (1.4%)	0.00	-0.02	-0.01
test	2051	47 (2.3%)	-0.01	-0.04	-0.03

Besides, this analysis shows that around more than three quarters of these errors are due to keyphrases incorrectly labelled as Material (43%) or Process (42%), whereas only 15% are Task. Interestingly, a similar proportion of keyphrases is observed in the training data: there is a considerably lower number of keyphrases labelled as Task (1132), than Process (2992) and Material (2608). For example, *nuclear fission reactors* in S0263822312000657 was labelled as Material but it is a Task in the gold standard data; *capture features in the solution* (S0021999113006955) was predicted as Task but it should be a Process; or *optimized coefficients* in S0021999113002945 was predicted as a Task but it is a Material.

Regarding coverage, 62 entities are not covered by System 2 at all. This amounts to 35% of the gold standard data. The distribution of errors is very similar to the one reported for precision, with 45% of the entities not covered being Material, 40% Process and 15% Task. For instance, in S0021999113006955 there are two instances of *true surface* that were ignored by the classifier. Interestingly, another mention of the same keyphrase in the same document was correctly annotated as Material. However, postprocessing of predictions to enforce consistent labelling in System 1 did not show any nett improvements.

References

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 the International Workshop on Semantic Evaluation*. Association for Computational Linguistics, Vancouver, Canada.

Biswanath Barik and Erwin Marsi. 2017. NTNU-2 at SemEval-2017 Task 10: Identifying Synonym and Hyponym Relations among Keyphrases in Scientific Documents. In *Proceedings of the International Workshop on Semantic Evaluation*. Vancouver, Canada.

Matthew Honnibal and Mark Johnson. 2015. An improved non-monotonic transition system for dependency parsing. In *EMNLP*. Lisbon, Portugal, pages 1373–1378.

John Lafferty, Andrew McCallum, Fernando Pereira, et al. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the eighteenth international conference on machine learning, ICML*. volume 1, pages 282–289.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity linking meets word sense disambiguation: a unified approach. *TACL* 2:231–244.

Naoaki Okazaki. 2007. CRFsuite: a fast implementation of Conditional Random Fields (CRFs). http://www.chokkan.org/software/crfsuite/.