Proceedings of the 6th International Workshop on Computational Terminology (COMPUTERM 2020), pages 95–100
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

TermEval 2020: TALN-LS2N System for Automatic Term Extraction

Amir Hazem, Mérième Bouhandi, Florian Boudin, and Béatrice Daille
LS2N - UMR CNRS 6004, Université de Nantes, France
{amir.hazem,merieme.bouhandi,florian.boudin,beatrice.daille}@ls2n.fr

Abstract

Automatic terminology extraction is a notoriously difficult task aiming to ease effort demanded to manually identify terms in domain-specific corpora by automatically providing a ranked list of candidate terms. The main ways that addressed this task can be ranged in four main categories: (i) rule-based approaches, (ii) feature-based approaches, (iii) context-based approaches, and (iv) hybrid approaches. For this first TermEval shared task, we explore a feature-based approach, and a deep neural network multitask approach -BERT- that we fine-tune for term extraction. We show that BERT models (RoBERTa for English and CamemBERT for French) outperform other systems for French and English languages.

Keywords: Terminology extraction, Feature-based, BERT.

1. Introduction

Automatic terminology extraction (ATE) is a very challenging task beneficial to a broad range of natural language processing applications, including machine translation, bilingual lexicon induction, thesauri construction (Lin, 1998; Wu and Zhou, 2003; van der Plas and Tiedemann, 2006; Hagiwara, 2008; Andrade et al., 2013; Rigouts Terryn et al., 2019), to cite a few.

Traditionally, this task is conducted by a terminologist, but hand-operated exploration, indexation, and maintenance of domain-specific corpora and terminologies is a costly enterprise. The automatization aims to ease effort demanded to manually identify terms in domain-specific corpora by automatically providing a ranked list of candidate terms.

Despite being a well-established research domain for decades, NLP methods still fail to meet human standards, and ATE is still considered an unsolved problem with considerable room for improvement. If it is generally admitted that terms are single words or multiword expressions representing domain-specific concepts and that terminologies are the body of terms used with a particular domain, the lack of annotated data and agreement between researchers make ATE evaluation very difficult (Terryn et al., 2018). In order to gather researchers around a common evaluation scheme, TermEval shared task (Rigouts Terryn et al., 2019) offers a unified framework aiming a better ATE's comprehension and analysis [1]. The shared task provides four data sets: Corruption, dressage, wind energy and heart failure; in three languages: English, French and Dutch.

With the advance of neural network language models and following the current trend and excellent results obtained by transformer architecture on other NLP tasks, we have decided to experiment and compare two classification methods, one feature-based and the BERT-based. We show that BERT models (RoBERTa for English and CamemBERT for French) outperform other systems for French and English languages. Also, the feature-based approach shows competitive results.

2. Task Description

The shared task provides four data sets. Three of them are dedicated to the training phase: corruption, dressage and wind energy, and one to the test phase: heart failure. All the corpora are provided in three languages: English, French and Dutch. The data sets are described in detail in (Rigouts Terryn et al., 2019). Five teams have participated in the TermEval shared task. All teams submitted results for English, three submitted for French and two for Dutch. We submitted results for the French and English data sets. The Precision, recall, and F1-score were calculated twice: once including and once excluding Named Entities.

3. Proposed System

We present in this section the two experimented approaches during the training phase that is: (i) the feature-based and, (ii) the BERT-based approaches. For the test phase, the submitted results are those of BERT approach only. However, we also report the obtained results of the feature-based approach for comparison.

3.1. Feature-based Approach

3.1.1. Feature Extraction

Classical methods for extracting terms from corpora often consist of three major steps: the first one uses some linguistic filtering, the second one consists of describing the candidates through different features in order to give them a weight indicating the degree of confidence that they are indeed a term, and the third is more of a selection phase. As for the first step, we know that, often, the first requirement is for a term to be a noun phrase, and our main morphosyntactic pattern is defined (primarily by observing recurrent patterns in the given reference lists of terms): a noun or nouns (or proper nouns), which might be preceded or followed by adjectives (vertical axis wind turbine), or of-genitives (United States of America). These patterns are then passed to spaCy's rule-matching engine[2] to extract a list of candidate terms. Once our candidate terms are extracted, we processed to the second step, and we assign to

[1] https://termeval.ugent.be/

[2] https://spacy.io/

each one of them linguistic, stylistic, statistic, and distributional descriptors that might help us get insights as to the nature of terms (Table 1). In this work, beyond the common statistical descriptors, we wanted to focus on different measures of specificity and termhood, since we know that a term is much more common and essential in a specialized corpus than it is in a general domain corpus. Termhood is defined by (Kageura and Umino, 1996) as ”the degree to which a linguistic unit is related to domain-specific context”:

- Measures of specificity and termhood
  - Specificity ($Specificity$): $Specificity(a) = 2 \cdot \frac{f_D(a) \times f_G(a)}{f_D(a)+f_G(a)}$ with a the term, $f_D(a)$ the term frequency in the specialized corpus and $f_G(a)$ its out-of-domain frequency.
  - Term's relation to Context (W_{rel}): $W_{rel}(a) = (0.5 + ((WL \cdot \frac{TF(a)}{MaxTF}) + PL)) + (0.5 + ((WR \cdot \frac{TF(a)}{MaxTF}) + PR))$ with $TF(a)$ the term frequency in the document, $MaxTF$ the frequency of the most frequent word, WL (or [WR]) is the ratio between the number of different words that co-occur with the candidate term (on the left [right] side) and the total number of words that it co-occurs with. PL (or [PR]) is the ratio between the number of different words that co-occur with the candidate term (on the left [right] side) and the $MaxTF$. W_{rel} measures the singularity of the term a in the corpus and quantifies the extent to which a term resembles the characteristics of a stopword. The more a candidate word co-occurs with different words, the more likely it is to be unimportant in the document.
  - Cvalue ($Cval$): $Cval(a) = log_2|a| \cdot (f(a) - \frac{1}{P(T_a)} \sum_{n \in T_a} f(n))$ with $f(a)$ the frequency of term a, $|a|$ the number of words in a, T_a is the set of extracted candidate terms that contain a, $P(T_a)$ is the total number of longer candidate terms that contain a.
  - Termhood (TH): $W(a) = \frac{f_a^2}{n} \cdot \sum_1^n (log \frac{f_{n,D}}{N_D} - \frac{f_{n,R}}{N_R})$ with f_a^2 the absolute frequency of the word in the domain-specific corpus, n the number of words in a, $\frac{f_{n,D}}{N_D}$ the frequency of each constituant of the term in the domain-specific corpus ($\frac{f_{n,R}}{N_R}$ for the general domain) relative of the size of the corpora (in tokens).

As for the last step, classification is conducted to select the terms using these features.

3.1.2. Classification

Boosting is a classification method that consists of iteratively learning several classifiers whose individual weights are corrected as they go along to better predict difficult values. The classifiers are then weighted according to their performance and aggregated iteratively. We use the XGBoost model (eXtreme Gradient Boosting) (Chen and Guestrin, 2016), and we feed it our feature vectors after being normalized using *sklearn*[3] standard scaler, which transforms an x value into a $z = \frac{x-u}{s}$ value, with u being the mean of the x and s its standard deviation. However, these features can be more or less essential to characterize our terms. After several tests, we have empirically determined that only the elements that correlate at more than a certain threshold (mean correlation) with our target class are retained for classification (bolded in 1).

Feature	Reference
First letter is a capital letter	-
Number of words	-
Length of term in characters	-
Number of stopwords	-
Relevance (how many other candidates contain this term)	-
Position of the first occurrence	(Aquino et al., 2014)
Spread	(Hasan and Ng, 2014)
TF, IDF, TF-IDF	(Jones, 2004)
Relative frequency (RF, in and out-of-domain)	-
Sum of subparts' RF (in and out-of-domain)	-
Specificity (harmonic mean of RF in-domain and RF out-domain)	-
Cvalue	(Vu et al., 2008)
Z-Score	(Aquino et al., 2014)
Term's relation to context	(Campos et al., 2018)
Termhood	(Vintar, 2010)

Table 1: Summary Table of Features

3.2. BERT

BERT has proven to be efficient in many downstream NLP tasks (Devlin et al., 2018) including next sentence prediction, question answering and named entity recognition (NER). It can also be used for feature extraction or classification. Prior to the emergence of transformer-based architectures like BERT, several deep learning architectures for terminology extraction have been proposed. Wang et al. (2016) introduce a weakly-supervised classification-based approach. Amjadian et al. (2016) leverage local and global embeddings to encapsulate the meaning and behavior of the term for the classification step, although they only work with unigram terms.

We must note that exploring these architectures is not the focus of this work; we mainly want to observe how BERT-based models can be used for ATE and how they perform in comparison to more traditional feature-based methods. In order to do that, we use different versions of BERT as a binary classifier for term prediction.

For English, we use RoBERTa (Liu et al., 2019), which is a model built based on BERT but modifies key hyperparameters in the original BERT model, eliminating its next-sentence pretraining objective and training the model with much larger mini-batches and more substantial learning rates, leading to more solid downstream task performance. For French, we use CamemBERT (Martin et al., 2019), the French version of the BERT model. For both languages, we

[3] https://scikit-learn.org/stable/

	English																	
	NES									ANN								
Tools	Corp			Equi			Wind			Corp			Equi			Wind		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Patterns	2.42	**76.2**	5.60	6.60	68.4	11.7	1.50	76.1	2.40	2.60	**61.5**	5.10	6.80	50.2	10.7	1.20	55.8	2.20
Features	**40.6**	16.4	23.7	**38.7**	19.4	25.5	**51.1**	10.9	17.2	**39.4**	17.6	24.4	**38.7**	19.1	25.4	**51.2**	10.8	17.4
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
BERT3	27.1	41.4	32.8	28.4	82.0	**42.2**	22.2	81.1	**34.8**	18.4	35.6	24.2	20.5	80.6	**32.7**	16.0	82.4	**26.9**
BERT4	28.5	38.9	32.9	26.5	**85.0**	40.4	21.3	83.8	33.9	17.8	30.7	22.5	19.1	**83.5**	31.1	15.4	85.6	26.2
BERT5	25.5	42.9	32.0	27.3	80.5	40.8	19.9	**93.5**	32.8	16.7	35.7	22.8	19.4	78.1	31.1	14.7	90.5	25.4
BERT6	25.6	57.6	**35.5**	27.6	84.3	41.6	16.9	89.9	28.5	18.7	53.4	**27.7**	19.8	82.6	32.0	12.4	**93.0**	22.0
	French																	
	NES									ANN								
Tools	Corp			Equi			Wind			Corp			Equi			Wind		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Patterns	3.08	**74.6**	5.93	5.26	67.9	9.76	1.69	77.2	3.31	3.69	**72.8**	7.08	6.75	71.3	12.3	2.09	76.3	4.08
Features	30.9	25.1	27.7	**54.3**	11.5	19.6	**46.4**	16.4	24.2	**31.5**	25.9	28.4	**54.3**	11.5	19.1	**45.4**	16.4	**24.1**
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
BERT3	41.5	23.1	29.6	25.8	61.7	**36.4**	18.2	58.9	27.8	23.2	15.9	18.9	19.8	58.1	**29.5**	13.6	55.5	21.9
BERT4	27.9	48.5	35.4	24.8	63.0	35.6	17.9	67.4	**28.2**	20.8	44.9	28.4	18.9	59.1	28.7	13.7	64.8	22.6
BERT5	30.1	57.2	39.5	20.1	71.2	31.4	11.3	76.9	19.8	23.1	54.4	32.5	15.7	68.6	25.6	8.89	75.2	15.9
BERT6	**36.7**	48.4	**41.7**	9.08	**78.1**	16.2	9.11	**82.5**	16.4	26.6	43.5	**33.1**	7.27	**76.8**	13.2	7.21	**81.8**	13.2

Table 2: Terminology extraction scores (%) obtained on the training data sets. BERT3 for instance, stands for BERT using ngrams of length 3 for training.

will use pre-trained models, and both of them are fine-tuned during the classification. The general objectives BERT is trained on gives the model an innate sentence classification capability. The main idea is to associate each term with its context. Hence, by analogy to the next sentence prediction, the first sentence given to BERT is the one which contains the term, and the sentence to predict is the term itself. For training, we feed the model with all the context/term pairs that appear in the corpus as positive examples. The negative examples are generated randomly. Given the following sentence: ”*this is the first global instrument in the fight against corruption*”, *corruption* is annotated as a positive example (term) and a randomly chosen word or n-gram, *global* for instance, is annotated as a negative example. It is important to highlight the fact that the negative examples are all the n-grams that do not appear in the training evaluation term list. Also, the number of negative examples is equal to the number of positive ones.

4. Experiments and Results

Hand-engineering features is a challenging assignment, even more so for a task as challenging as extracting terms from domain-specific corpora and finding features to capture the right characteristics for each term and stay relevant with any corpora in hand. We can observe, from our results (table 2), that we often fail to find a good trade-off between recall and precision. As a matter of fact, with features as strict as these, we often find ourselves with correct precision and quite a weak recall.

4.1. BERT Settings

For the fine-tuning phase of BERT, we used the simple-transformers [4] library and its default parameters setting. For English, we used RoBERTa with n-gram size of four while for French, we used CamemBERT with n-gram size of five.

4.2. Experiments on the Training Data Sets

We started with the hypothesis that the features of a term noun phrase must be different from the features of a non-term noun phrase and that the features that characterize these terms must be valid from one corpus to the other. However, we can clearly see that the main problem encountered with the feature-based method is that the features learned by the model are hardly transferable from one corpus to another, as the notion of the relevance of each candidate term changes from one application area to another, and from one domain to another. Hard-coded features learned on one corpus do not transfer well to another during classification, since not only are the texts and domains vary greatly, but even the range of the values for the noun phrases features in the different corpora can vary enormously (see figures 1, 2, 3). Going for a feature-less method seems to be a nice direction to explore (table 4). Our experiments with BERT, even if they were somewhat successful, were a bit abrupt, since we consider all the n-

[4] `https://github.com/ThilinaRajapakse/simpletransformers`

grams as potential candidates, without prior filtering. We end up, after classification, with false positives in our list, such as phrases beginning or ending with pronouns or conjunctions. One of the reasons that pushed us first to test this configuration without prior noun phrases filtering was our fear of losing potential positive candidates (we can in table 4 see that recall post-filtering is average). Future work will incorporate syntactic information into this BERT process in order to get better precision.

4.3. Results of the Test Set

	English					
	NES			ANN		
	P	R	F1	P	R	F1
TALN-LS2N	34.78	**70.87**	**46.66**	32.58	**72.68**	**44.99**
RACAI	42.40	40.27	41.31	38.57	40.11	39.33
NYU	**43.46**	23.64	30.62	**42.18**	25.12	31.48
e-Termino	34.43	14.20	20.10	34.43	15.54	21.42
NLPLab	21.45	15.59	18.06	20.06	15.97	17.78
	French					
	NES			ANN		
	P	R	F1	P	R	F1
TALN-LS2N	**45.17**	**51.55**	**48.15**	**41.88**	**50.88**	**45.94**
e-Termino	36.33	13.50	19.68	36.33	14.37	20.59
NLPLab	16.07	11.18	13.19	15.12	11.20	12.87

Table 3: Official results on the heart failure test set(%).

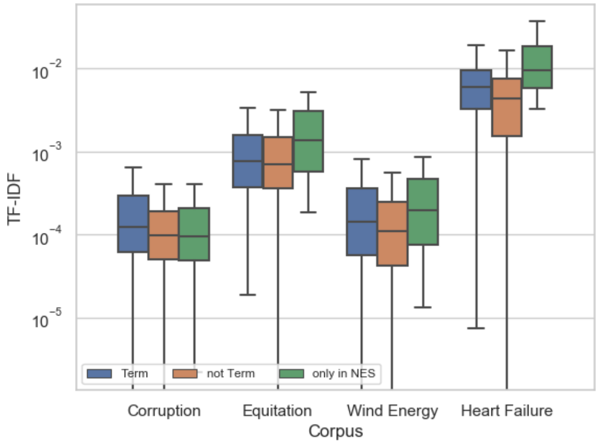

Figure 1: Range of the TFIDF values on all the corpora for English

	English					
	NES			ANN		
	P	R	F1	P	R	F1
Patterns	11.8	77.3	20.5	12.9	77.1	22.1
Features	**39.4**	29.2	33.6	**39.6**	29.4	33.7
	P	R	F1	P	R	F1
BERT3	34.0	69.9	**45.7**	31.5	70.9	**43.6**
BERT4	31.7	78.3	45.2	29.3	79.1	42.7
BERT5	26.9	**83.7**	40.8	24.9	**84.6**	38.4
BERT6	30.8	77.7	44.1	28.3	78.3	41.6
	French					
	NES			ANN		
	P	R	F1	P	R	F1
Patterns	16.9	65.3	25.8	17.9	65.1	27.8
Features	**48.9**	53.4	**50.9**	**48.9**	53.3	**50.9**
	P	R	F1	P	R	F1
BERT3	41.3	58.5	48.4	38.5	58.0	46.3
BERT4	40.2	66.9	50.3	37.7	66.8	48.2
BERT5	34.3	73.1	46.7	32.2	73.2	44.7
BERT6	24.3	**76.3**	36.9	22.9	**76.4**	35.2

Table 4: Results on the heart failure test set(%) using BERT with different ngram's size. BERT3 for instance, stands for BERT using ngrams of length 3 for training.

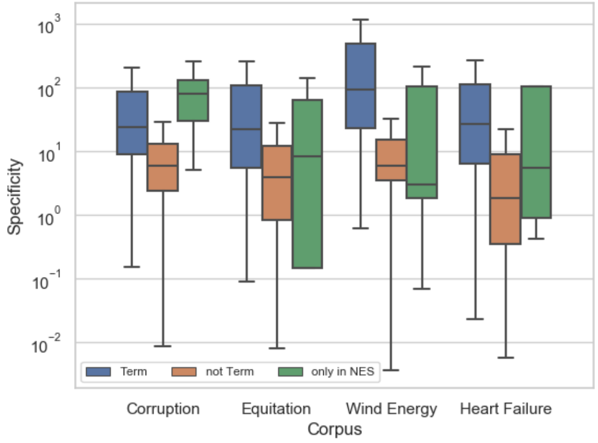

Figure 2: Range of the Specificity values on all the corpora for English

The results on the test set are consistent with the results on the training corpora. The same patterns can be observed, and results on the test set are in the same range. Based on the F1-score, our approach represented by TALN-LS2N using BERT obtained the best results of the competition. However, we see that in terms of precision, the NYU team obtained the best results for English. Overall, feature-based and BERT-based approaches exhibit similar performance on the French test set while for English, BERT is more accurate. Further experiments are certainly needed to improve both methods. However, the capability of BERT (certainly thanks to its attention mechanism) to learn hidden features suggests less effort is needed compared to the feature-based approach, which requires more efforts in the design of the features. Also, the n-gram size used in BERT was fixed empirically based on the development data sets. Further anal-

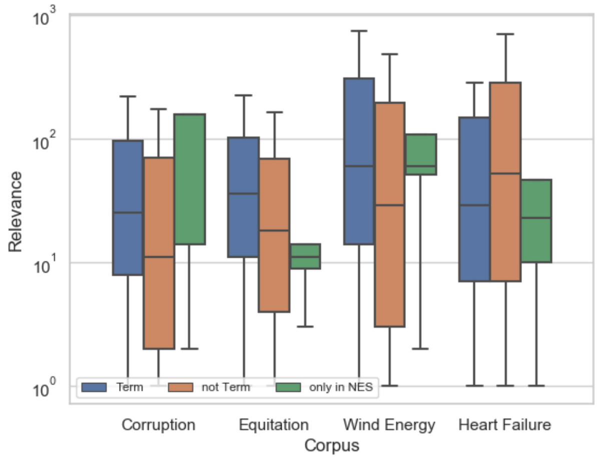

Figure 3: Range of the Relevance values on all the corpora for English

ysis is needed to make our approach n-gram independent for better term length coverage. Indeed, we limited our system outputs to 4-grams for English and 5-grams for French, which did not allow the extraction of longer terms. Finally, recent work has shown several improvements of BERT such as StructBERT (Wang et al., 2020) and T5 (Raffel et al., 2019). These recent state-of-the-art approaches can, in the future, be used to further improve the results of ATE.

5. Conclusion

Term extraction has been a very active field of research for many decades. Methods based solely on linguistic analysis and patterns have given way to new statistical, machine, and deep learning methods. We conducted several experiments using classical hand-engineered features-based methods in order to find the best way to extract terms in several specialized domains. These models that combine linguistic, statistical and distributional descriptors suggest that the relation between test and training corpora are of central importance. Moreover, we have seen that it is only natural for the very notion of termhood in different domains to be more pragmatic than theoretical. We then proposed a BERT-based classification approach that outperformed classical methods on this shared task. This contribution is setting a new, simple and strong baseline for terminology extraction. However, the overall results of this task are average at best, and much room is left for improvement.

6. Bibliographical References

Amjadian, E., Inkpen, D., Paribakht, T., and Faez, F. (2016). Local-global vectors to improve unigram terminology extraction. In *Proceedings of the 5th International Workshop on Computational Terminology (Computerm2016)*, pages 2–11, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Andrade, D., Tsuchida, M., Onishi, T., and Ishikawa, K. (2013). Synonym acquisition using bilingual comparable corpora. In *International Joint Conference on Natural Language Processing (IJCNLP'13)*, Nagoya, Japan.

Aquino, G., Hasperué, W., and Lanzarini, L. (2014). Keyword extraction using auto-associative neural networks.

Campos, R., Mangaravite, V., Pasquali, A., Jorge, A. M., Nunes, C., and Jatowt, A. (2018). A text feature based automatic keyword extraction method for single documents. In Gabriella Pasi, et al., editors, *Advances in Information Retrieval*, pages 684–691, Cham. Springer International Publishing.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *CoRR*, abs/1603.02754.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Hagiwara, M. (2008). A supervised learning approach to automatic synonym identification based on distributional features. In *Proceedings of the ACL-08: HLT Student Research Workshop*, pages 1–6, Columbus, Ohio, June. Association for Computational Linguistics.

Hasan, K. S. and Ng, V. (2014). Automatic keyphrase extraction: A survey of the state of the art. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1262–1273, Baltimore, Maryland, June. Association for Computational Linguistics.

Jones, K. S. (2004). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 60(5):493–502.

Kageura, K. and Umino, B. (1996). Methods of automatic term recognition: A review. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 3(2):259–289.

Lin, D. (1998). Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 2*, ACL '98, pages 768–774, Stroudsburg, PA, USA. Association for Computational Linguistics.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Martin, L., Muller, B., Ortiz Suárez, P. J., Dupont, Y., Romary, L., Villemonte de la Clergerie, É., Seddah, D., and Sagot, B. (2019). CamemBERT: a Tasty French Language Model. *arXiv e-prints*, page arXiv:1911.03894, Nov.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2019). Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv e-prints*.

Rigouts Terryn, A., Hoste, V., and Lefever, E. (2019). In no uncertain terms: a dataset for monolingual and multilingual automatic term extraction from comparable corpora. In *Language Resources and Evaluation*.

Terryn, A. R., Hoste, V., and Lefever, E. (2018). A Gold Standard for Multilingual Automatic Term Extraction from Comparable Corpora: Term Structure and Translation Equivalents. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 7-12, 2018.

European Language Resources Association (ELRA).

van der Plas, L. and Tiedemann, J. (2006). Finding synonyms using automatic word alignment and measures of distributional similarity. In *21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics ACL'06*, Sydney, Australia.

Vintar, S. (2010). Bilingual term recognition revisited: The bag-of-equivalents term alignment approach and its evaluation. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 16(2):141–158.

Vu, T., Aw, A., and Zhang, M. (2008). Term extraction through unithood and termhood unification. In *IJCNLP*.

Wang, R., Liu, W., and McDonald, C. (2016). Featureless domain-specific term extraction with minimal labelled data. In *Proceedings of the Australasian Language Technology Association Workshop 2016*, pages 103–112, Melbourne, Australia, December.

Wang, W., Bi, B., Yan, M., Wu, C., Xia, J., Bao, Z., Peng, L., and Si, L. (2020). Structbert: Incorporating language structures into pre-training for deep language understanding. In *International Conference on Learning Representations*.

Wu, H. and Zhou, M. (2003). Optimizing synonym extraction using monolingual and bilingual resources. In *In Proceedings of the second international workshop on Paraphrasing*, page 72.