TermInteract: An Online Tool for Terminologists Aimed at Providing Terminology Quality Metrics

Pedro Hernández-Vegas
Artificial Intelligence Department
Universidad Politecnica de Madrid
Madrid, Spain
p.hvegas@upm.es

Lucía Guasp
Artificial Intelligence Department
Universidad Politecnica de Madrid
Madrid, Spain
l.guasp@upm.es

Mariano Rico
Artificial Intelligence Department
Universidad Politecnica de Madrid
Madrid, Spain
mariano.rico@upm.es

2020 6th IEEE Congress on Information Science and Technology (CiSt) | 978-1-7281-6646-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/CiST49399.2021.9357197

***Abstract*—This position paper describes the results of our meetings with terminologists aimed at capturing missing functionalities concerning term extraction. These experts lack an online tool to allow them the evaluation, modification and creation of terminologies from corpora in different languages. The result is an open-access online website where terminologists can extract automatically terminologies and can upload terminologies created by third parties. The tool also allows them to polish collaboratively the terminologies. Additionally, this tool provides a quality metrics based on the changes made to the terminology, as well as a way to compare two terminologies for the same corpus but created with different methods. We have tested the tool with terminologies in English and Spanish, with corpora from several domains: bio (covid19), legal and scientific papers. Our preliminary results point out the utility of this tool for terminologists.**

***Index Terms*—terminologies, pattern-based part of speech, neural networks, corpora**

I. Introduction

Terminology extraction is increasingly becoming a fundamental sub-task in the information extraction process of Natural Language Processing (NLP). In recent years, numerous techniques, based on supervised approaches [1]–[3] as well as unsupervised approaches [4]–[6], have been developed and improved, providing increasingly better tools for experts and researchers, but some problems and shortcomings remain.

Despite all the tools and systems that have been developed in the literature, experts in the field still need functionalities that have not been effectively addressed. The most complete systems, such as sketchengine.eu[1] has limitations like the lack of an effective terminology comparator and the lack of terminology quality metrics.

We contacted with Spanish terminologists belonging to the Spanish Association of Terminology[2] and had several meetings aimed at identifying these missing functionalities. From these meetings we obtained a requirements specification that we have implemented in the online tool that we describe in this position paper.

This tool has been created to cope with tasks like the generation of terminologies, their edition and most importantly, their comparison. Here we also present some of the results of the comparison of terminologies generated from very different methods such as methods based on regular expressions (unsupervised) and methods based on neural networks (supervised).

For the sake of reproducibility, we have published all the information related to this paper in the following public site: `https://keyq.linkeddata.es`. The datasets and source code used to extract terminologies with these two different approaches can be found in this site. Regarding data, we have made available the three different corpus used in our experimental setup. Additionally, the results obtained with the experiments and the link to the tool created is also available on the site.

The rest of the paper is organized as follows. Section II explains the methods that have been used to extract terminologies, section III shows the methodology used by the terminologists. In section IV we show the experiments that have been carried out in this work. Section V details the results obtained with the development of the experiments and finally, section VI presents the conclusions and future work.

II. Methods for creating terminologies

Term extraction is a process that consists of obtaining the keywords or expressions from a text automatically. There are plenty of methods for terminology extraction in the state of the art. Nowadays the more classical unsupervised techniques for terminology extraction, such as the ones based on regular expressions [4], Support Vector Machines (SVM) [5], or Naïve Bayes models [6] [7], coexist with supervised learning techniques such as sequential tagging models [1], models based on decision trees [2] or neural networks and deep learning [3] methods. These neural methods are known to achieve the best results.

A. Pattern-based Part of Speech (POS) method

This method based on regular expressions is characterized by the use of syntactic patterns underlying the morpho-syntactic structure of a text. In order to extract terms from a corpus using regular expressions, it is necessary to tag all the words of the corpus. This grammatical tagging process, also

We would like to thank Accenture for his partial support under the project KeyQ, hosted at the AInnovation center in UPM Montegacedo campus (`http://ainnovation.upm.es`).

[1] See `https://www.sketchengine.eu/`

[2] See Asociación Española de Terminología (Spanish Association of Terminology, `https://aeter.org/terminesp/`).

© IEEE 2021. This article is free to access and download, along with rights for full text and data mining, re-use and analysis.

known as lexical disambiguation or POS (Part Of Speech) tagging is a challenge for which many NLP experts have devoted much attention, developing powerful tools, especially for English. The tagging process, which is quite intensive in computational terms, attempts to classify words by writing them down in a grammatical category or word classes. The labels in which the words can be categorized can also vary depending on the labeling system used. We distinguish here between (1) methods that use SPOS (simple POS) tagging with 8 word classes such as noun (N) or adjective (A) and (2) methods that use UPOS (universal dependencies) tagging with 17 word classes such as proper noun (PROPN) or a pronoun (PRON).

1) SPOS patterns: Once the text has been tagged with the morpho-syntactic information, we have to find out regular expressions [8] that retrieve the terms. For instance, the equation 1 is a regular expression to retrieve Simple Noun Phrases (SNP) for English using SPOS word classes.

$$(A|N)^*N(P+D^*(A|N)^*N)^* \tag{1}$$

where:

A is an adjective but not a determinant
N is a noun.
P is a preposition.
D is a determinant.

2) universal dependencies UPOS patterns: This UPOS pattern-based approach is based on the same principles as the SPOS pattern-based approach, except for the labels used to perform the pattern identification. While the SPOS pattern-based methods has a set of 8 word classes, the UPOS pattern-based method employs the universal dependencies, which provide a greater degree of labeling with 17 word classes. Therefore, this feature allows us a better specification of the desired word types.

B. Neural methods

Many different algorithms of machine learning models may be applied to the task of automatic terminology extraction. This task can be solved as a sequence labeling problem [7]. In these models, each of the tokens in the texts is classified as belonging to a term or not belonging to it (binary classification [2] [9]) or as a multi-class classification [10], distinguishing whether a word starts a term, belongs to it without being the first word of the term, or is not part of any term.

Different neural structures have been used in the literature, varying in size, network topology and configuration. In terms of the type of neural network layer, as in most natural language processing tasks, recurrent networks tend to be used and more specifically, bi-directional LSTM networks, as they are the ones that best relate the context of the input sequences.

In this type of models, word embeddings have been introduced, such as Glove [1] or Word2Vec [11], in order to improve the way models used to symbolise words. Language models have also been used in the literature to represent words and solve term extraction tasks. These context-based representations of words (such as ELMo [12], OpenAI GPT [13] or BERT [3]) provide better results compared to fixed representations (Glove and Word2Vec) [10] but we could not reproduce the cited results. Therefore,we followed the work of Basaldella et al. [14], in which it is used a neural model whose architecture is based on bidirectional LSTM cells and the word embeddings of Glove. We could reproduce this structure and has been used in the experiments of this work, named as NN1.

Additionally, another model (named NN2) has been developed with a deeper architecture, with two hidden layers of type BiLSTM, which allows the model to learn specific characteristics of the input data. Additionally, the Adam optimizer has been used [15], which combines the techniques of RMSProp and Momentum, having demonstrated a better performance in the scientific literature. The structure of this model is represented in Figure 1.

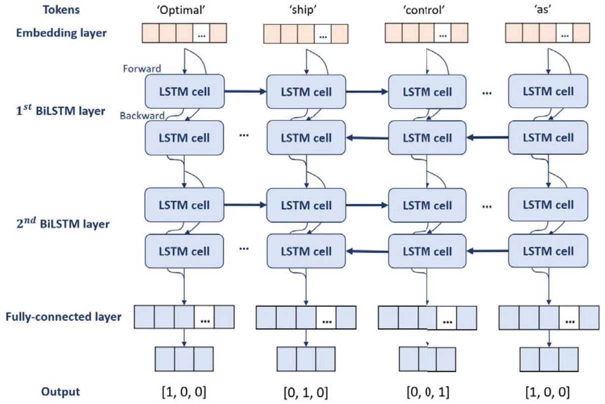


Fig. 1. Representation of the structure of model NN2 aimed at extracting terminologies.

III. Terminologist methodology

In our meetings with terminologists, besides functionalities we obtained the methodology they would follow when using their ideal tool. Specifically, what methods would be used to (1) evaluate the quality of a given terminology and (2) compare two given terminologies. For the first aspect, a metrics based on the number of required editions was agreed, following the current state of the art. However, for the second aspect, a procedure new to the literature was envisaged. This procedure to compare two terminologies is based on some agreements among the terminologists: (1) a criteria for sorting the terms and (2) the number of terms to evaluate.

With these two parameters set in the tool, each terminologists get a list of terms that must evaluate for each terminology (as shown in figure 2). The terminologist will review each term in both lists in the following way: if a term is valid there is no action but, if a term is wrong (any non perfect term is considered as wrong) a simple click on the term will increase the counter of wrong terms. Once both list are reviewed, the number of items identified as wrong is a quality metrics for comparing these two terminologies.

The terminologists did not agreed on a specific method for sorting the terms. Therefore, we provide them with three methods: TF-IDF, RAKE, CVALUE. This way, terminologists can agree on one of these or, alternatively, if there is no agreement, they evaluate both terminologies using the three criteria and the average mean of these three methods is computed. This average score selects which terminology has better terms.

The sorting methods are described as follows:

CValue. This statistical method [16], and its derivatives like NCValue [17], is mainly used in domains with a large number of technical terms. To achieve this, CValue scores the extracted terms according to whether they are contained in some other candidate terms or not, as shown in equation 2.

$$CValue(a) = \begin{cases} \log|a| \cdot f(a) & \text{if } a \text{ not cont.} \\ \log|a| \cdot (f(a) - \frac{1}{T(a)} \sum_{b \in Ta} f(b)) & \text{if } a \text{ contained} \end{cases} \quad (2)$$

Where a is the candidate term being scored, $f(a)$ is the total frequency of occurrences of a in the study corpus, $T(a)$ is the set of candidates containing a within it, and $f(b)$ is the total frequency of occurrences of b, containing a, in the study corpus.

TF-IDF. This method (Text Frequency - Inverse Document Frequency) [8] scores each term according to the documents in which it appears, as shown in equation 3.

$$\begin{aligned} tf(t,d) &= \frac{freq(t,d)}{\sum_i^a freq(t_i,d)} \\ idf(t) &= \ln(\frac{N}{n(t)}) \end{aligned} \quad (3)$$

Where $tf(t,d)$ is the relative frequency of a term t in a given document d, $freq(t,d)$ are the occurrences of a term t in a given document, $idf(t)$ is the inverse document frequency of a term t, N is the total number of documents in the study corpus, and $n(t)$ is the number of documents in the corpus containing the term t.

RAKE. Rapid Keyword Extraction is a term extraction method in itself [18] which incorporates a scoring method of its own, stands out for its speed. This scoring method favors longer terms over terms that are composed of fewer words.

IV. Experiments

A. Description of the corpora used

Here we present a brief description of the corpora used in the experiments shown in this paper. More details on these datasets can be found in the website of the project.

- Covid-19 corpus. With a size of 1.2GB, this tar.gz file contains 681 pdf files retrieved on 2020/03/20 using the Kaggle challenge metadata[3]. Each pdf is a scientific paper in English. It comprises 8,405 pages and 7,294,168 tokens.
- INSPECT corpus [19]. It comprises 2,000 scientific abstracts on TIC, computers and control, manually annotated (identifying the terms). One thousand documents were used for training, 500 for testing and 500 for validating. These abstracts have a range between 15 and 557 words (average 125 words). The average number of terms in each abstract is 15.

B. POS methods

Although the tool allows us to specify any regular expression, like the ones used in [4], for our experiments we have used the SNP pattern [8] (see equation 1). The SNP patterns are positioned as the best alternative due to the lack of other patterns in English created for the fields treated by the chosen corpora.

This choice means that in addition to a comparison between different methods, we can deal with a comparison between an unsupervised method, such as pattern-based POS, and another supervised method like the neural approaches.

Obtaining the terminologies using regular expressions is a very simple task using our tool. The pipeline used to generate these terminologies is identical for both corpora under study, and it is composed of the following steps:

First, it is necessary to tag the text, and more specifically, all the words that appear in it. This tagging task (also known as disambiguation task) tries to classify the words by noting them in a grammatical category or word classes. The tags in which the words can be categorized may also vary depending on the tagging system used.

There are different tagging systems, which have characteristics that make them more suitable in different cases of use, such as the EAGLE tagging [20] system or the UPOS tagging [21] system. These taggers are powerful tools based on manually tagged corpus that are known as Treebanks.

Once the words in a corpus have been tagged, the system find out the patterns in the text, that is, matches the patterns in the regular expression provided.

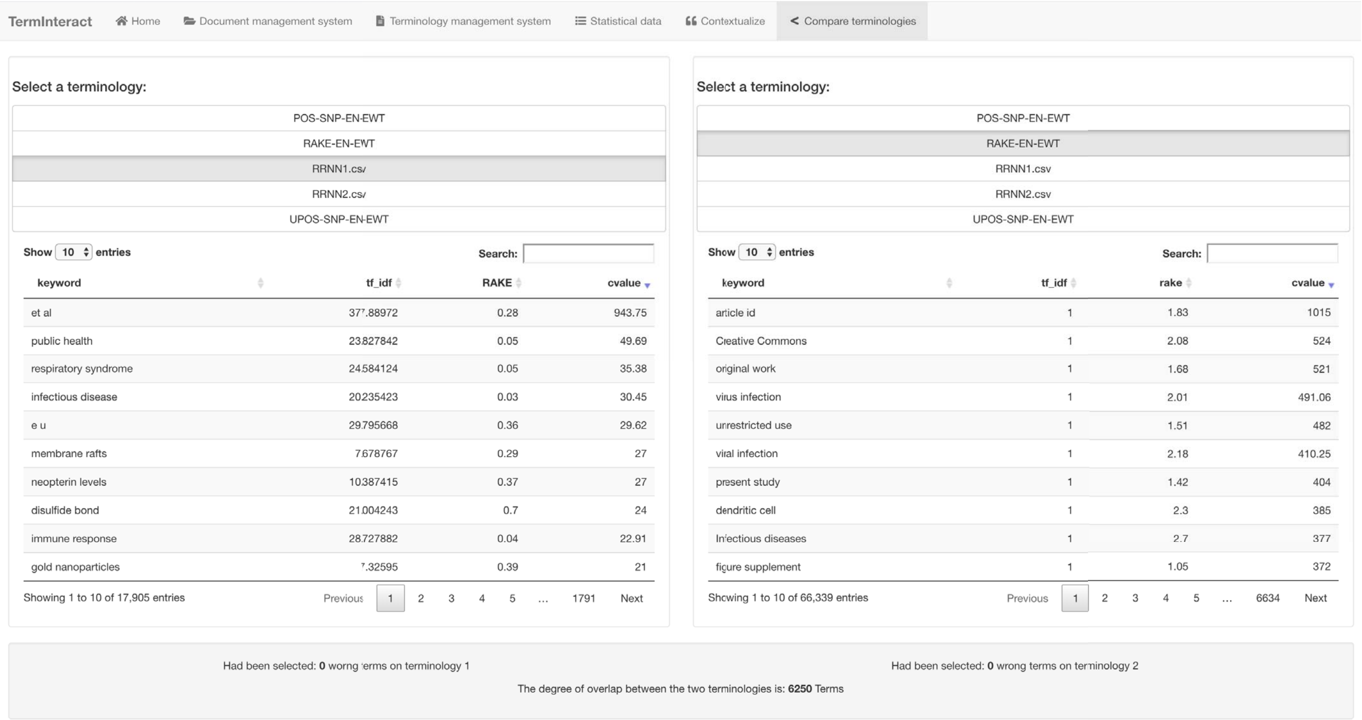
As last step, the tool also allows us to classify the words extracted by sorting under different techniques such as TF-IDF [8] or CVALUE [16].

C. Neural methods

Once the structure of the neural network has been defined, the model must be trained with a labeled dataset, which contains an annotated terminology selected by experts. The public dataset used in this work to train, test and validate the neural methods has been the INSPEC dataset [19]. This training process allows the model to define the parameters of the neural network that allows to reduce the cost function defined in the algorithm, thus obtaining a valid model for terminology extraction. The categorical crossentropy cost function has been used, as this is a multi-class classification task.

To prevent the network from becoming too close to the training data and overfitting, several regularization techniques have been used, such as L2, which reduces the degrees of freedom of the model by penalizing the weights of the neuron

[3] See `https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge`.



Select a terminology:

POS-SNP-EN-EWT
RAKE-EN-EWT
RRNN1.csv
RRNN2.csv
UPOS-SNP-EN-EWT

Show 10 entries Search:

keyword	tf_idf	RAKE	cvalue
et al	377.88972	0.28	943.75
public health	23.827842	0.05	49.69
respiratory syndrome	24.584124	0.05	35.38
infectious disease	20.235423	0.03	30.45
e u	29.795668	0.36	29.62
membrane rafts	7.678767	0.29	27
neopterin levels	10.387415	0.37	27
disulfide bond	21.004243	0.7	24
immune response	28.727882	0.04	22.91
gold nanoparticles	7.32595	0.39	21

Showing 1 to 10 of 17,905 entries

Select a terminology:

POS-SNP-EN-EWT
RAKE-EN-EWT
RRNN1.csv
RRNN2.csv
UPOS-SNP-EN-EWT

Show 10 entries Search:

keyword	tf_idf	rake	cvalue
article id	1	1.83	1015
Creative Commons	1	2.08	524
original work	1	1.68	521
virus infection	1	2.01	491.06
unrestricted use	1	1.51	482
viral infection	1	2.18	410.25
present study	1	1.42	404
dendritic cell	1	2.3	385
Infectious diseases	1	2.7	377
figure supplement	1	1.05	372

Showing 1 to 10 of 66,339 entries

Had been selected: **0** worng terms on terminology 1

Had been selected: **0** wrong terms on terminology 2

The degree of overlap between the two terminologies is: **6250** Terms

Fig. 2. Section of the tool for comparing terminologies. The user can select which terms are not correct, and the system counts them, providing a comparison metrics between the two terminologies. In this case the terminologists agreed on sorting the terms by cvalue.

connections, and Dropout, with a value of 25%, which means that at each step of the training, each neuron has a 25% probability of being deactivated.

The performance of the two neural models has been measured on the test data set, following the most common metrics of accuracy, recall and F1. The model NN2 allows to obtain a value of F1 metrics with a confidence interval between 46.51% and 47.43% with a 95% of confidence level. These results have shown that similar performance of the state of the art is achieved. After verifying the functioning of the models, they have been used to extract the terminology associated with the COVID-19 corpus, whose results will be discussed in the following section.

V. Analysis of results

The results obtained with our tool are as expected. That methods based on neural networks give the best results was a known fact, but this tool follows the methodology of the terminologists and provides numbers for comparing terminologies.

In this section we present two tables with results extracted from the tool. Table I presents the results for the 'INSPEC' corpus, while table II presents the results for the 'Covid-19' corpus.

The evaluated methods POS, UPOS, RAKE and the two neural network based structures (NN1 and NN2) are heading the columns in both tables. These tables also show information like the language models used and the patterns used for term extraction. Information is also provided on the number of terms extracted in each method, and the frequency ranges presented by the extracted terms. Due to the characteristics of each method it can be seen that the neural networks provide terminologies with terms of frequency 0 in table I. This fact has been solved in the second case (covid corpus), table II, where the 0-frequency terms have been eliminated.

The TF-IDF, RAKE score and CVALUE columns indicate the number of terms identified as misleading for the first 100 terms when ordered using these criteria. Therefore, terminologies with lower scores are more adequate. Notice how the RAKE extraction method does not applies (NA) for classification through TF-IDF due to the loss of information that occurs in this process. In table II, the NAs that appear in the UPOS extraction of RAKE and CVALUE scores are due to execution time longer than 9 days. Therefore, it has not been possible to obtain conclusive results.

Finally, in each table there is a column with the score, composed of the arithmetic mean of the scores obtained with the other classification methods.

VI. Conclusions and future work

Our objective has been to create an online tool useful for managing terminologies with the functionalities that terminologists consider necessary and which do not exist in the literature, as far as we know.

We have tested this tool comparing terminologies obtained with different methods based on two different approaches: (1) unsupervised, using POS regular expressions and (2) supervised, using neural networks.

Method	SPOS	UPOS	RAKE	NN1	NN2
POS Model	english-ewt	english-ewt	english-ewt	NA	NA
POS pattern	(A|N)*N(P+D*(A|N)*N)*	((ADJ|NUM)|(NOUN|PROPN|PRON))* (NOUN|PROPN|PRON)(ADP+DET* ((ADJ|NUM)|(NOUN|PROPN|PRON))* (NOUN|PROPN|PRON))*	NOUN ADJ	MODEL1	MODEL2
#Terms	28.089	20.727	1.914	7.253	7.794
Frequencies	1 - 279	1 - 279	2 - 123	0 - 88	0 - 87
Evaluation (#terms removed)					
TF-IDF score	11	9	NA	1	1
RAKE score	15	12	3	4	3
CVALUE score	2	2	0	0	0
Average score	9,33	7,67	1,5	1,6	1,3

TABLE I

Results for the ‘INSPECT’ corpus. The lower the score in each ranking method, the better the terminology, as the score indicates how many wrong terms have been found in the first 100 terms.

Method	SPOS	UPOS	RAKE	NN1	NN2
POS Model	english-ewt	english-ewt	english-ewt	NA	NA
POS Pattern	(A|N)*N(P+D*(A|N)*N)*	((ADJ|NUM)|(NOUN|PROPN|PRON))* (NOUN|PROPN|PRON)(ADP+DET* ((ADJ|NUM)|(NOUN|PROPN|PRON))* (NOUN|PROPN|PRON))*	NOUN ADJ	MODEL1	MODEL2
#Terms	1.797.817	1.246.380	66.339	17.905	14.021
Frequencies	1 - 20.839	1 - 20.839	2 - 20.308	1 - 5.785	1 - 2.966
Evaluation (#terms removed)					
TF-IDF score	8	8	NA	4	3
RAKE score	100	NA	30	14	9
CVALUE score	44	NA	1	4	1
Average score	50,67	NA	15,5	7,3	4,33

TABLE II

Results fro the ’Covid-19’ corpus. Notice the high number of terms identified. Due to the long execution times (more than one week), conclusive results for the UPOS method are not obtained yet. Therefore, the results for this method are not shown.

The comparison between the terminologies obtained with these approaches confirm what literature point out: models based on neural network architectures achieve better results. Furthermore, with this tool now we can have a precise metrics of these differences. Besides, this tool has other functionalities such as (1) automatic generation of terminologies based on POS patterns, (2) collaborative edition of terminologies, not shown in this position paper, and (3) terminology comparison based on criteria pointed out by terminologists (TF-IDF, CVALUE and RAKE).

There are various lines of future work that will be carried out to complete and improve this work. We will explore how to combine the terms obtained from different terminology extraction methods. For instance, pattern-based POS methods may generate terms that neural methods do not, and vice versa. The degree of overlapping will be checked and we will study different methods to complement the results of these methods to get an even better terminology.

Another line of future work will provide training sets to the pattern-based POS systems in order to have a supervised system, as it is the case for the neural method. This line will check whether the results achieved by these (both supervised) methods are so significant. Additionally, the tool will be extended to measure the degree of agreement among terminologists using metrics like Fleiss’ Kappa [22] or Krippendorff’s Alpha [23].

References

[1] R. Alzaidy, C. Caragea, and C. L. Giles, “Bi-LSTM-CRF Sequence Labeling for Keyphrase xtraction from Scholarly Documents,” in *The world wide web conference*, 2019, pp. 2551–2557.

[2] P. D. Turney, “Learning Algorithms for Keyphrase Extraction,” *Information retrieval*, vol. 2, no. 4, pp. 303–336, 2000.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” *arXiv preprint arXiv:1810.04805*, 2018.

[4] M. Rico, P. Calleja, P. Martın, and E. Montiel, “Extracting Terminologies in the Legal Domain: a Syntactic Pattern-based Approach for Spanish,” in *Iberlegal workshop at JURIX conference*, 2019.

[5] W. Ni, T. Liu, and Q. Zeng, “Extracting Keyphrase Set with High Diversity and Coverage Using Structural SVM,” in *Asia-Pacific Web Conference*. Springer, 2012, pp. 122–133.

[6] I. H. Witten, G. W. Paynter, E. Frank, C. Gutwin, and C. G. Nevill-Manning, “Kea: Practical automated keyphrase extraction,” in *Design and Usability of Digital Libraries: Case Studies in the Asia Pacific*. IGI global, 2005, pp. 129–152.

[7] S. D. Gollapalli and X.-l. Li, “Keyphrase Extraction Using Sequential Labeling,” *arXiv preprint arXiv:1608.00329*, 2016.

[8] J. S. Justeson and S. M. Katz, “Technical terminology: some linguistic properties and an algorithm for identification in text,” *Natural language engineering*, vol. 1, no. 1, pp. 9–27, 1995.

[9] E. Frank, G. W. Paynter, I. H. Witten, C. Gutwin, and C. G. Nevill-Manning, “Domain-specific keyphrase extraction, to appear in: Proceedings of the sixteenth international joint conference on artificial intelligence,” 1999.

[10] D. Sahrawat, D. Mahata, M. Kulkarni, H. Zhang, R. Gosangi, A. Stent, A. Sharma, Y. Kumar, R. R. Shah, and R. Zimmermann, “Keyphrase Extraction from Scholarly Articles as Sequence Labeling using Contextualized Embeddings,” *arXiv preprint arXiv:1910.08840*, 2019.

[11] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed Representations of Words and Phrases and Their Compositionality,” in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[12] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” *arXiv preprint arXiv:1802.05365*, 2018.

[13] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving Language Understanding by Generative Pre-training,” 2018.

[14] M. Basaldella, E. Antolli, G. Serra, and C. Tasso, “Bidirectional lstm recurrent neural network for keyphrase extraction,” in *Italian Research Conference on Digital Libraries*. Springer, 2018, pp. 180–187.

[15] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[16] S. Ananiadou, “A methodology for automatic term recognition,” in *COLING 1994 Volume 2: The 15th International Conference on Computational Linguistics*, 1994.

[17] K. Frantzi, S. Ananiadou, and H. Mima, “Automatic recognition of multi-word terms:. the c-value/nc-value method,” *International journal on digital libraries*, vol. 3, no. 2, pp. 115–130, 2000.

[18] S. J. Rose, W. E. Cowley, V. L. Crow, and N. O. Cramer, “Rapid automatic keyword extraction for information retrieval and analysis,” 6 2012, uS Patent 8,131,735.

[19] A. Hulth, “Improved automatic keyword extraction given more linguistic knowledge,” in *Proceedings of the 2003 conference on Empirical methods in natural language processing*. Association for Computational Linguistics, 2003, pp. 216–223.

[20] B. Baldwin, C. Doran, J. C. Reynar, M. Niv, B. Srinivas, and M. Wasson, “Eagle: An extensible architecture for general linguistic engineering.” in *ANLP*, 1997, p. 23.

[21] S. Petrov, D. Das, and R. McDonald, “A universal part-of-speech tagset,” *arXiv preprint arXiv:1104.2086*, 2011.

[22] J. L. Fleiss and J. Cohen, “The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability,” *Educational and psychological measurement*, vol. 33, no. 3, 1973.

[23] K. Krippendorff, “Estimating the reliability, systematic error and random error of interval data,” *Educational and Psychological Measurement*, vol. 30, no. 1, 1970.