Benchmark for Automatic Keyword Extraction in Spanish: Datasets and Methods

Pablo Calleja[1], Patricia Martín-Chozas[1] and Elena Montiel-Ponsoda[1]

[1]Ontology Engineering Group, Universidad Politécnica de Madrid

Abstract

Tasks such as document indexing or information retrieval still seem to heavily rely on keywords, even in the LLMs era. However, there is still a need for automatic keyword extraction works and training sets in languages other than English. To the best of our knowledge, no datasets for keyword extraction in Spanish are publicly available for training or evaluation purposes. Additionally, those innovative keyword extraction methods that rely on language models are not being adapted to language models in other languages. To palliate this situation, this work proposes a method to translate into Spanish two of the main gold standard datasets used by the community, while preserving semantics and terms. Then, the main state-of-the-art methods are evaluated against the new translated datasets. The methods used for the evaluation have been configured or re-implemented for Spanish.

Keywords

Spanish Automatic Keyword Extraction, Spanish language, SemEval2017, SemEval2010

1. Introduction

Keywords, typically defined as words or terms that best characterise the topics discussed in a document, have proven essential for different NLP tasks such as information extraction (IE), text mining, or information retrieval (IR) [1]. With the exponential growth of available digital documents, a need emerged for algorithms capable of automatically identifying single or compound terms (also referred to as key segments or key phrases) that best represented the most relevant information of a document, a task better known as Automatic Keyword or KeyPhrase Extraction (AKE).

Nowadays, even in the face of AI Generative algorithms and Large Language Models (LLMs), AKE algorithms are not only used to classify, retrieve, or inspect large corpora [1, 2, 3], but also to fine-tune LLMs and post-process their output.

However, automatically extracting keywords is a challenging task due to the complexities of natural language, document heterogeneity and the type of keywords that usually are needed. The current state-of-the-art is full of proposed methods and tools. From the earliest based on lexico-syntactic patterns and frequencies [4] to those purely based on statistics [5, 6] or the most recent ones based on language models.

Keyword extraction methods have generally been classified into supervised or unsupervised methods. Traditional supervised methods are based on decision trees [7], naive Bayes [8] or Conditional Random Fields [9]. In the past 10 years, several models have emerged based on neural networks and deep learning [10, 11]. The most recent approaches rely on language models and attention mechanisms [12, 13].

Supervised methods tend to offer the best results in the literature of machine learning, but they require a large dataset of labelled training corpora. To achieve that, human experts have to manually annotate large amounts of data, which is a costly and tedious task. The resulting annotations refer to the specific keywords that should be extracted from each sentence, paragraph or document in the corpus. On the other hand, unsupervised methods, such as statistical or graph-based approaches, do not require labelled corpora. Statistical-based methods [5, 14] use candidate position, frequency, length, and capitalisation to determine the importance of a word. Graph-based approaches [15, 16] construct a graph with the candidates as nodes. The edges indicate similarity or co-occurrence of candidates.

Some of the best-known datasets for automatic keyword extraction such as SemEval2010 [17], SemEval2017 [18] or Inspec [19], have been created for evaluation tasks and are commonly used to evaluate new methods (both supervised and unsupervised), and not so much for training.

However, all these efforts are not language agnostic. Most of the works so far have been oriented towards the English language, giving a small coverage to other languages such as Spanish. As far as we know, there are no publicly available annotated training corpora in Spanish. Therefore, supervised algorithms cannot be easily implemented, and evaluations for supervised or unsupervised algorithms are difficult to perform.

SEPLN-2024: 40th Conference of the Spanish Society for Natural Language Processing. Valladolid, Spain. 24-27 September 2024.

p.calleja@upm.es (P. Calleja); patricia.martin@upm.es (P. Martín-Chozas); elena.montiel@upm.es (E. Montiel-Ponsoda)

0000-0001-8423-8240 (P. Calleja); 0000-0002-8922-7521 (P. Martín-Chozas); 0000-0003-3263-3403 (E. Montiel-Ponsoda)

In this paper, a method to translate two of the most important corpora for AKE is proposed and applied to their translation into Spanish. The main aim of this work is to create a 'silver standard ´ to support the training and evaluation of automatic keyword extraction in Spanish. The translation process has been performed to preserve the semantics and terminological representation of the original texts and the annotations. The translation is supported by the Google Translate service and by ChatGPT3.5.

Additionally, a benchmark has been generated with five of the most relevant methods in the current state-of-the-art on the two translated corpus. The methods have been configured for Spanish, and two of them have been re-implemented to use Spanish language models.

The rest of the paper is structured as follows: In section 2 we provide a summary of the state-of-the-art in Automatic Keyword Extraction. Section 3 is devoted to the method for the translation of the corpora. Section 4 describes the different AKE methods with their configurations or adaptations for the Spanish language, and section 5 presents the results of the evaluation benchmark. Finally, section 6 highlights the conclusions and recommendations for future work. Both experiments and results are reported in an anonymised GitHub repository[1].

2. State of the art

As stated by [1], 'keywords' and 'keyphrases' do not refer to any theory. An element is considered as a 'key' element within a document, when it is an important descriptor of the document content. The use of 'word' versus 'phrase' refers to the number of textual units, which can be one (1-gram) or several (n-grams). Since such keywords or keyphrases mostly correspond to terms, defined as words that are specific to a domain, the AKE task is closely related to the so-called Automatic Terminology Extraction/Retrieval (ATE/ATR) task, i.e., the task of identifying relevant terms in a corpus [20].

Lossio-Ventura et al. [21] described in their work that there are some fundamental differences between term extraction and keyword extraction tasks. One major difference is that extracting terms requires a large collection of texts, which is not a necessary requirement in keyword extraction, which can take only a single document as input. Also, ATE methods aim to extract term-like units and remove those that may not be terms, syntactically or terminologically. On the other hand, AKE methods extract the 'key' elements of a document, which are not limited to terms. Thus, while AKE methods can be domain independent, ATE methods apply to specific fields or professional domains, since their main goal is to build resources that contain the lexical units that are representative of a domain.

Although these two tasks have been conceived for different purposes, the truth is that, when performed automatically, they obtain similar results and performance, as both rely on linguistic and textual features (at sentence, paragraph or document levels). Thus, several state-of-the-art methods have been used for both tasks.

In this section, we will review the most relevant works in this area, making a distinction between traditional approaches (linguistic and statistic) and machine learning and neural approaches.

2.1. Traditional approaches

The algorithms considered in this section are usually based on linguistic patterns, relying on parsing and part-speech tagging processes to identify terms [22]. These patterns were very prolific in the 1990s, with systems such as LEXTER [23]. This kind of approaches [24] has persisted until today, as patterns are the main starting point to automatically identify keywords or terms in documents and corpora. More advanced works based on patterns went further to identify the concept evoked by term variants in several languages, as the work by [25] for English and French. In any case, the majority of these works are language dependent.

Later on, researchers started to combine various types of linguistic techniques, such as pattern-based techniques, regular expressions, stop word lists, and post-processing algorithms, to mention but a few. In this context, tools such as TermExtractor emerge, a system that combines several of the previously mentioned techniques and applies post-processing filters like domain pertinence, lexical cohesion or structural relevance [26].

More advanced works in the literature started to use statistical approaches in combination with linguistic functionalities, which appeared to improve the results. The process behind statistical approaches generally consists of weighting the frequency of occurrence of a combination of words (n-grams) in a text. Normally, statistical algorithms are divided into two types: 1) those based on the *unithood* that measures the strength of unity of complex units (such as X^2, T-score and z-score), and 2) those based on the *termhood* that measures the degree of representation of domain-specific concepts, such as C-Value or co-occurrence [27, 28]. Some of these purely statistical term extractors are INDEX for English [29], Lexterm [30] for Spanish, and RAKE [5], for keyword extraction in English.

In contrast, it is most common to find mixed approaches, such as TerMine, a term extractor that combines C-Value with linguistic information [4], or TermSuite, which applies distributional and compositional methods [31]. In [32], authors combine linguistic

[1] https://github.com/oeg-upm/spanish-termex

processes such as segmentation, PoS tagging and morphological analysis, with semantic knowledge extracted from external resources and statistical techniques. Other works, such as TextRank [33], create a graph from the text to extract keywords based on statistical metrics.

2.2. Machine Learning and Neural approaches

These approaches exploit different features (linguistic or not) to identify keywords. For instance, Rose et al. [5] identified keywords based on word frequency, the number of co-occurring neighbors, and the ratio between the co-occurrence and the frequency. Campos et al. [34] proposed YAKE which calculated the importance of each candidate using frequency, offsets, and co-occurrence. SemCluster method [35] first clustered the candidates based on the semantic similarity in which the centroids were selected as keywords. TopicRank [36] first assigned a score to each topic by candidate keywords clustering. The topics were scored using the TextRank ranking model, and keywords were extracted using the most representative candidate from the top-ranked topics. Florescu et al. [37] proposed PositionRank to use the position of word occurrences to improve TextRank on a document.

Word embeddings have also been widely used. Wang et al. [38] made use of the pre-trained word embedding and the frequency of each word to generate weighted edges between words in a document. A weighted PageRank algorithm was used to compute the final scores of words. Also, Key2Vec [39] used a similar approach using the phrase embeddings for representing the candidates and ranking the importance of the phrases by calculating the semantic similarity and co-occurrences of the phrases.

Currently, new approaches based on pre-trained neural language models have appeared in the literature. For instance, Text2TCS[2] [40], which is able to extract terms and relations from raw text, creating taxonomies automatically. [41] proposed SIFRank, the integration of a statistical model and a pre-trained language model, to calculate the relevance between candidates and document topics. Other works are focused on the extraction of multilingual terminology across domains using transformers [42].

Two of the most recent works in the field of AKE using language models are AttentionRank and MDERank. AttentionRank [13] integrates self-attention weights extracted from a pre-trained language model with the calculated cross-attention relevancy value to identify keywords that are important to the local sentence context and also have strong relevancy to all sentences within the whole document. MDERank [12] bases the identification of keywords on the embedding representation of the sentence using masked tokens. Moreover, their work proposes a new type of BERT architecture to be trained as a language model, but for the purpose of keyword identification.

[2]https://live.european-language-grid.eu/catalogue/tool-service/8122

3. Dataset generation

In the era of machine learning approaches, datasets are an essential requirement to train and, what is more important, evaluate algorithms for different NLP tasks. For instance, in the field of Automatic Keyword Extraction, there are well-known gold standard datasets that are commonly used to evaluate approaches within the literature such as the SemEval2010 Task 5 [17] and SemEval2017 Task 10 [18]. However, the availability of these data sets is limited to languages other than English [43]. Consequently, a common approach to overcome this limitation is to translate the available datasets into the target language [44, 45], including Spanish [46].

To the best of our knowledge, there is no consolidated dataset in Spanish for Automated Keyword Extraction, therefore, the first contribution of this work is the development of an evaluation corpus for keyword extraction in Spanish which results from translating two of the most common English AKE datasets: SemEval2010 and SemEval2017. The target of this contribution is to generate a 'silver standard' labelled dataset, to provide researchers in the field with a consolidated framework to test and evaluate their approaches.

However, the translation process for labelled datasets is not a straightforward task. As [47] demonstrated in their work, labelled datasets have their labels linked to one token or a span of tokens. Since the sentence structure can vary in different languages, it is very challenging to retain the same annotation structure after the translation process. To overcome such difficulties, we have organised the translation process into two phases: Phase 1) Source Dataset Analysis and Source Dataset Preprocessing, described in Section 3.1, and Phase 2) Source Dataset Translation and Target Dataset Postprocessing, described in Section 3.2.

Figure 1 summarises the method for the translation process in which, given the two original datasets, a set of four datasets translated into Spanish is obtained, using two different translation systems.

3.1. Phase 1: Dataset analysis and preprocessing

In order to generate the proposed silver standard for Spanish AKE, we have selected the two previously mentioned datasets, as they are widely used in experiments of this kind: SemEval2010 Task 5 [17] and SemEval2017

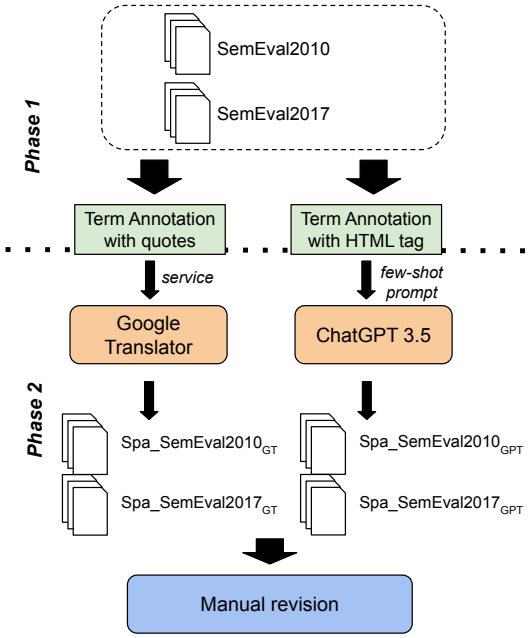

Figure 1: Method for dataset translation

Task 10 [18]. Both datasets are published following the same structure, a set of documents containing the raw text (named *docsutf8*) and a set of documents containing the extracted keywords (named *keys*). Both types of documents present the same identifiers to match keywords with source documents.

Despite their similar structure, they present several differences. As shown in Table 1, the main difference lies in their size. With a smaller number of documents, SemEval2010 far exceeds SemEval2017 in the total number of tokens, which means that it contains fewer documents, but of a much larger size. SemEval2017 contains shorter documents with an average of 6 to 7 sentences, whereas SemEval2010 contains full scientific papers with hundreds of sentences. It is interesting to note that, although SemEval2010 is bigger in number of documents and number of tokens, SemEval2017 has a bigger number of extracted keywords. This means that the keywords from SemEval2010 have greater representation and number of occurrences than the keywords from 2017. These differences in size are important because they require a different treatment of the documents during the preprocessing and the translation stage.

In both datasets, over 50% of the keywords are unigram or bigram. However, in SemEval2010 we observe that 555 keywords are not present in the documents with a similar span text. The reason for this is to be found in the way in which the original dataset was created. In SemEval2010,

Table 1
Metrics for SemEval2010 and SemEval2017 datasets, including keywords.

	SemEval2010	SemEval2017
Documents	243	493
Tokens	2.334.613	95.877
Keywords	3.785	8.529
Unmatched Keywords	555	0

some of the keywords come from the ones manually provided by the authors of the papers themselves, and they may not have an exact correspondence in the text.

Regarding the preprocessing of the datasets, there are two main aspects involved in the translation process. The first one refers to the original text. Not many issues were found during the translation of SemEval2017 corpus, since it had a manageable size and a clean structure. However, the original texts of SemEval2010 were arbitrarily segmented, very long, and contained references and formulas, which posed many problems for the automatic translator when processing them.

The second aspect refers to the keywords. For the translation of the keywords, we did not simply translate the list of keywords out of context, but decided to mark them in the texts with annotations marks (quotation marks or the HTML tag
, depending on the translation system). Then, we translated the texts and retrieved the translated terms contained within the annotation marks.

3.2. Phase 2: Dataset translation and postprocessing

Most of the existing approaches to create silver standards from existing gold standards by leveraging machine translation rely on at least two translation sources: one from a common online translator such as DeepL[3] or Google Translate[4], and the other using a Neural Machine Translation model, as suggested in [44]. As already announced, in this work we have used Google Translate and ChatGPT 3.5 Turbo[5] APIs.

The keywords from the texts that were translated with Google Translate were annotated with quotation marks. However, on some occasions the system retrieved errors in which the annotation marks were missing or misplaced in the translated sentence, and either it was not possible to extract the translated term from the annotated sentence or the extracted term was not correct. To avoid

[3]https://www.deepl.com/es/translator
[4]https://translate.google.es/
[5]https://platform.openai.com/docs/models/gpt-3-5-turbo

that, we decided to append the original term to each annotated sentence, to force the system to take that term into account and provide a translation. For instance, in the translation of the sentence '...has held two "mobile computing" design competitions' focused on the term 'mobile computing' the translation lost the quotation marks: 'ha celebrado dos concursos de diseño de computación móvil'. Thus, we add the term repeated to obtain the translation of the term: '...has held two "mobile computing" design competitions. Mobile computing'.

With ChatGPT, the tag
was used to mark the keywords before and after. The prompt sent to the generative model described the purpose of the model (i.e., 'You are a Spanish translator specialised in terminology'), and then some examples of annotations in English and its translations in Spanish with the annotated and translated keywords were provided. This is called few-shot prompting. The full prompt is presented in Annex A.

Regarding the postprocessing stage, several actions were performed. First, we extracted all the annotated occurrences of each keyword in the sentence, creating a list of translation candidates per keyword. In some cases, reconciliation between candidates was necessary to provide a single translation for each keyword. In the case that no disparities between the candidates were found, the translated keyword was automatically assigned. In case of disparities, terms were manually reviewed and a translated keyword was manually assigned. In total, we manually reviewed an average number of 2000 keywords per dataset (220 documents in SemEval2010 and 360 documents in SemEval2017).

4. AKE Adaptation to Spanish

In this section, the different AKE methods used for the experiments and their implementation are presented. Some of them have already been implemented and maintained by well-known Python libraries and contain adapters to work with other languages. Two of them, those that are based on language models, had to be re-implemented and adapted. In addition to different technical aspects, both methods use the original BERT model [48] for English, and the RoBERTa MarIA model [49] for Spanish.

4.1. Already implemented methods

The methods used for the evaluation are TopicRank, YAKE and RAKE. The Python library PKE[6] has been used for the execution of the TopicRank and YAKE methods. PKE uses the Python library spaCy[7], as many other methods, to identify candidate chunks or nominal phrases that can be relevant terms or keywords. Thus, the Spanish model of spaCy has to be downloaded before the methods can be run.

For the RAKE method, the original library cannot be used as it is only oriented to the English language. However, there is a version named Multi-rake[8] which covers different languages. As the method is statistical, to perform multilingually, the addition of stopword lists from the different target languages is necessary.

4.2. Attention Rank

The implementation of the original authors[9] had to be reimplemented from scratch. The original repository does not have libraries and version specifications. Moreover, the original code relies on libraries for language models that are not maintained as well as the noun phrases identification component, which relies on the part-of-speech annotation of Stanford CoreNLP and a third-party library. Reproducibility was not possible in this work.

A new repository[10] has been created for the implementation of the Attention rank method. This repository uses HuggingFace's library transformer to manage language models and spaCy to identify noun phrases. The repository details the specific libraries and versions needed and the external modules needed. The new repository allows the use of BERT (as in the original work) and RoBERTa architecture models in different languages.

The adaptation for RoBERTa models had to deal with two specific issues regarding the tokeniser. The first one is the use of different special tokens to delimit sentences at the beginning and at the end to focus the attention mechanisms, as BERT uses '[CLS]' and '[SEP]' tokens, RoBERTa uses '<s>' and '</s>'. The second issue is the generated tokens, as BERT uses a WordPiece tokeniser in which subwords are marked with the '##' tag (e.g., the word *thicknesses* is divided into tokens *thickness* and *##es*). In contrast, RoBERTa models use Byte-level Pair Encoding (BPE) and classifies different tokens for char sequences that start a word or that are inside. The tokens that start a word include the white space before the word, and they are marked whith the special character 'Ġ'. For instance, the word *extrapolate* is divided into two tokens: 'Ġextrap' and 'olate'.

Beyond the differences studied in previous works on the benefits or differences between both types of tokenisers [50], this work had to develop the alignment process between the words of keywords and their corresponding tokens. With WordPiece is easier to find tokens and recompose the original word, but BPE is sensible to appearance of the white space before the token. If it does not appear, the token is different and its attention value

[6] https://github.com/boudinfl/pke
[7] https://spacy.io/
[8] https://github.com/vgrabovets/multi_rake
[9] https://github.com/hd10-iupui/AttentionRank
[10] https://github.com/oeg-upm/AttentionRankLib

changes. This issue has been solved by ensuring that the input sentences always have a white space before a word.

4.3. MDERank

The original implementation[11] contains a better description of the requirements. However, it is described for Python 3.7 which is no longer supported by the community and most of the versions of the required libraries are deprecated. Also, parts of the execution code are wrong such as the command line execution or the arguments, and there is no code related to the KPEBERT model, a model which is trained and used for keyword identification. Only it is possible to execute it with traditional BERT models.

To update the code and method, a new repository has been created[12]. In which the requirements, code and execution process have improved. As AttentionRank, MDERank used Stanford CoreNLP for the identification of noun fragments and it has been updated to spaCy. Finally, the method can now support RoBERTa models by taking into account the problems mentioned in AttentionRank.

5. Evaluation

This section discusses the evaluation results obtained from the execution of the five AKE methods on the four translated datasets (Spa_SemEval2010$_{GT}$, Spa_SemEval2010$_{GPT}$, Spa_SemEval2017$_{GT}$ and Spa_SemEval2017$_{GPT}$). The metrics used in the evaluation are precision, recall and f1-measure. Following previous works in the literature, the methods are evaluated with the three metrics at the top K of the keywords extracted in each method. K equals 5, 10, and 15. Finally, we perform an error analysis and present a discussion around it. Table 2 shows the results obtained.

5.1. Results

Table 2 shows the results for each top K (5, 10, 15) and method. The results have been grouped by the type of dataset and the translation system used: Spa SemEval2010$_{GT}$, Spa SemEval2010$_{GPT}$, Spa SemEval2017$_{GT}$ and Spa SemEval2017$_{GPT}$, where GT stands for Google Translate and GPT stands for ChatGPT 3.5. Additionally, the column named BR, that stands for Best Result, shows the best f1 result reported in the original datasets in English (BR$_{10}$ for SemEval2010 and BR$_{17}$ for SemEval2017). These results are taken from the AttentionRank work [13], except for the results for MDERank, which are taken from their own published work [12].

[11] https://github.com/LinhanZ/mderank
[12] https://github.com/oeg-upm/mderanklib

The results of the AKE algorithms on the Spanish datasets, both multilingual and adapted for Spanish, show a lower performance compared to the original datasets. However, they are in line with the results for English. Unlike many other NLP experiments, where a good result is represented by metrics starting at 0.6 or 0.7 of f1 score, the highest metrics achieved by the algorithms tested in SemEval2010 and 2017 do not exceed 0.3821 (BR$_{17}$ and K= 15).

We already expected lower values, as the translation process is not perfect and it is not always possible to maintain the correlation of one keyword in English to the same keyword in Spanish. Apart from the errors detected (explained in Section 5.2), GPT3 showed better performance in maintaining the structure and terminology of the translated document.

It is also important to mention the different results obtained for each dataset. For Spa SemEval2017$_{GT}$ and Spa SemEval2017$_{GPT}$ the best results, in terms of precision, recall and f1-score, are obtained by the two methods that are based on language models: AttentionRank and MDERank. Although the original dataset contains complex keywords, the language models perform well as in the English dataset.

Surprisingly, for Spa SemEval2010$_{GT}$ and Spa SemEval2010$_{GPT}$ the best results are obtained by YAKE. The nature of the documents in SemEval2010, which are full papers without any cleaning, including formulas, references and citations, makes it difficult for a language model to perform well. An added issue is the large length of the documents, which in the case of RAKE produces results close to zero.

5.2. Error Analysis and Discussion

After a thorough analysis of the results, we conclude that, beyond some translation errors, the main reason behind the low numbers seems to be the poor quality of some keywords in the original datasets. Although both datasets are claimed to have been either generated or reviewed by humans, we have detected a great number of anomalies that may be the main source of errors, as we try to illustrate below:

- Duplicated structures: We find similar structures with small variations which produce noise and inconsistencies, such as terms with determiners (i.e. *metal* and *the metal*), terms with symbols or special characters (i.e. *logical inference* and *"logical inference"*), and terms with different spellings (i.e *reputation mechanism* and *Reputation mechanism*).

- Misspelled structures: We found several examples of misspelled structures, and, specifically, missing

Table 2
Evaluation of five AKE methods against the translated datasets measuring Precision (p), Recall (r) and F-measure (F). Each evaluation has taken into account the K (n top) value for 5, 10 and 15. Also, the best F1 obtained for the original SemEval2010 and SemEval2017 in English (BR_{10} and BR_{17}) with each method is reported.

k	Method	Spa_SE2010_{GT}			Spa_SE2010_{GPT}			BR_{10}	Spa_SE2017_{GT}			Spa_SE2017_{GPT}			BR_{17}
		p	***r***	***F1***	***p***	***r***	***F1***	***F1***	***p***	***r***	***F1***	***p***	***r***	***F1***	***F1***
5	RAKE	0.00	0.00	0.00	0.08	0.03	0.04	0.67	12.17	3.97	5.98	14.88	5.15	7.66	13.24
	TopicRank	4.77	1.65	2.45	7.08	2.53	3.73	5.26	19.39	5.85	8.99	21.94	6.87	10.47	15.92
	YAKE	7.49	2.58	3.83	**10.95**	**3.85**	**5.69**	8.46	10.47	3.39	5.13	18.86	6.45	9.61	12.05
	AttentionRank	7.52	**2.60**	**3.86**	9.30	3.32	4.89	11.39	**19.51**	**5.88**	**9.03**	24.66	7.84	11.89	**23.59**
	MDERank	**7.63**	2.44	3.70	9.62	3.11	4.70	**12.95**	19.39	5.60	8.69	**27.46**	**7.94**	**12.32**	22.81
10	RAKE	0.00	0.00	0.00	0.16	0.11	0.13	1.33	12.70	8.16	9.93	14.86	10.07	12.00	22.61
	TopicRank	4.77	3.28	3.89	6.38	4.50	5.28	7.43	15.98	9.45	11.88	17.97	11.07	13.70	20.60
	YAKE	**7.37**	**5.07**	**6.01**	**9.42**	**6.56**	**7.74**	11.98	11.87	7.62	9.28	18.09	12.19	14.56	18.16
	AttentionRank	7.22	4.38	5.45	9.11	5.45	6.81	15.12	**16.71**	**9.96**	**12.48**	20.54	12.91	15.85	**34.37**
	MDERank	7.17	4.59	5.60	8.88	5.74	6.97	**17.07**	15.92	9.20	11.66	**22.45**	**12.98**	**16.45**	32.51
15	RAKE	0.05	0.05	0.05	0.11	0.11	0.11	1.78	11.98	11.25	11.60	14.02	13.90	13.96	26.87
	TopicRank	4.36	4.39	4.38	5.38	5.65	5.51	8.02	13.61	12.10	12.81	15.09	13.85	14.44	22.37
	YAKE	**6.83**	**7.02**	**6.93**	**8.56**	**9.04**	**8.79**	12.87	11.33	10.70	11.01	17.20	17.09	17.15	20.72
	AttentionRank	6.70	5.83	6.23	7.90	7.97	7.93	16.66	**14.20**	**12.52**	**13.31**	17.09	15.93	16.49	**38.21**
	MDERank	6.27	6.03	6.15	7.79	7.54	7.66	**20.09**	13.84	12.01	12.86	**19.31**	**16.75**	**17.93**	37.18

letters both at the beginning and at the end of the structure (i.e. *aked* instead of *baked*).

- Non-terminological structures: This is the most common anomaly in both datasets, and one of the main causes for the low performance of the algorithms, both in English and in Spanish. Examples of such non-terminological structures are: full sentences (i.e. *dynamics which clearly reveal the origins of the roaming*), sentence fragments (i.e. *loading force and penetration depth were recorded and their respective values were correlated with the observed*), concatenated structures (i.e.1. *well defined phase space dividing surfaces attached to*, i.e.2. *austenitic or austenitic & ferritic stainless steel*), or even text fragments with references (i.e.1. *comparison between the realistic calculations for positive parity [12] and negative parity [14], based on the same quark model [15]*, i.e.2. *calculation by Martinez-Pinedo et al.*).

Additionally to inaccuracies and anomalies mentioned before, in the results we observe that in some instances the same keyword has been translated differently into Spanish in different parts of the text. For example, the term *deployment* has been translated both as *despliegue* and *implementación* within the same text; or the compound term *information aggregation* can be found translated as *agregación de información* and *agregación de la información*. In itself, this would not be a problem because these are correct translations in Spanish. Moreover, even in specialised domains, term variants are commonly used to designate the same concept.

A similar issue occurs when Spanish terms vary in gender and number. For instance, the keyword *ferromagnetic* can be found translated into two different keywords throughout the text, as *ferromagnética* and *ferromagnéticos*. However, with the aim to be faithful to the original evaluation datasets, we decided to choose one of the translations and discard the alternatives, although we believe that the datasets would benefit from including such variation.

6. Conclusions

This work has analysed the current state-of-the-art of automatic keyword extraction and, in particular, the Spanish landscape. In this analysis, we have identified the lack of an evaluation framework (including datasets and ready-to-test algorithms) for AKE in Spanish. Consequently, this paper proposes two contributions. First, the generation of a silver standard for the Spanish language community by the translation of two English datasets widely used to evaluate AKE approaches: SemEval2010 and SemEval2017. Second, the configuration of a set of state-of-the-art algorithms in an easily executable manner to facilitate the evaluation task, including the adaptation of two current methods that rely on language models: Attention Rank and MDERank.

With the benchmark in place, we have performed an evaluation of the implemented algorithms and the translated datasets. To be consistent with the evaluations in English, the translated datasets maintain the original inner structure. The results in Spanish suggest the same tendency as in English, although they are lower. The error analysis shows that low results are due to several factors: 1) the quality of the original datasets, as they contain noisy texts, non-terminological structures, and

terms that are not contained in the texts, 2) the quality of the translations for the labelled datasets, as both systems present translation inconsistencies and have difficulties to keep track of the translated keyword in the text, 3) the fact that a 1 to 1 translation of keywords is not always possible nor desirable, and that it would be recommendable to include term variants.

In light of the results and taking these remarks into account, we conclude that maintaining the dataset structure in English to evaluate AKE tasks in Spanish might not be the most appropriate approach. For this reason, as part of future work we are considering two approaches for generating evaluation datasets in Spanish: 1) automatically postprocessing existing datasets, such as the two dealt with in this work, to eliminate all non-terminological structures and produce a list of candidate terms instead of just one in the translation process, and 2) semi-automatically generating a dataset with similar characteristics to the ones mentioned, but based on texts originally written in Spanish.

Acknowledgments

This work has been partially founded by INESDATA (https://inesdata-project.eu/) project, funded by the Spanish Ministry of Digital Transformation and Public Affairs and NextGenerationEU, in the framework of the UNICO I+D CLOUD Program - Real Decreto 959/2022.

References

[1] N. Firoozeh, A. Nazarenko, F. Alizon, B. Daille, Keyword extraction: Issues and methods, Natural Language Engineering 26 (2020) 259–291. doi:10.1017/S1351324919000457.

[2] O. Borisov, M. Aliannejadi, F. Crestani, Keyword extraction for improved document retrieval in conversational search, arXiv preprint arXiv:2109.05979 (2021).

[3] H. Shah, R. Mariescu-Istodor, P. Fränti, Webrank: Language-independent extraction of keywords from webpages, in: 2021 IEEE International Conference on Progress in Informatics and Computing (PIC), IEEE, 2021, pp. 184–192.

[4] K. Frantzi, S. Ananiadou, H. Mima, Automatic recognition of multi-word terms:. the c-value/nc-value method, International journal on digital libraries 3 (2000) 115–130.

[5] S. Rose, D. Engel, N. Cramer, W. Cowley, Automatic keyword extraction from individual documents, Text mining: applications and theory 1 (2010) 1–20.

[6] A. Oliver, M. Vàzquez, Tbxtools: a free, fast and flexible tool for automatic terminology extraction, in: Proceedings of the international conference recent advances in natural language processing, 2015, pp. 473–479.

[7] P. D. Turney, Learning algorithms for keyphrase extraction, Information retrieval 2 (2000) 303–336.

[8] I. H. Witten, G. W. Paynter, E. Frank, C. Gutwin, C. G. Nevill-Manning, Kea: Practical automatic keyphrase extraction, in: Proceedings of the fourth ACM conference on Digital libraries, 1999, pp. 254–255.

[9] D. Sahrawat, D. Mahata, M. Kulkarni, H. Zhang, R. Gosangi, A. Stent, A. Sharma, Y. Kumar, R. R. Shah, R. Zimmermann, Keyphrase extraction from scholarly articles as sequence labeling using contextualized embeddings, arXiv preprint arXiv:1910.08840 (2019).

[10] R. Alzaidy, C. Caragea, C. L. Giles, Bi-lstm-crf sequence labeling for keyphrase extraction from scholarly documents, in: The world wide web conference, 2019, pp. 2551–2557.

[11] R. Meng, S. Zhao, S. Han, D. He, P. Brusilovsky, Y. Chi, Deep keyphrase generation, arXiv preprint arXiv:1704.06879 (2017).

[12] L. Zhang, Q. Chen, W. Wang, C. Deng, S. Zhang, B. Li, W. Wang, X. Cao, Mderank: A masked document embedding rank approach for unsupervised keyphrase extraction, arXiv preprint arXiv:2110.06651 (2021).

[13] H. Ding, X. Luo, Attentionrank: Unsupervised keyphrase extraction using self and cross attentions, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, 2021, pp. 1919–1928.

[14] R. Campos, V. Mangaravite, A. Pasquali, A. M. Jorge, C. Nunes, A. Jatowt, Yake! collection-independent automatic keyword extractor, in: Advances in Information Retrieval: 40th European Conference on IR Research, ECIR 2018, Grenoble, France, March 26-29, 2018, Proceedings 40, Springer, 2018, pp. 806–810.

[15] X. Wan, J. Xiao, Single document keyphrase extraction using neighborhood knowledge., in: AAAI, volume 8, 2008, pp. 855–860.

[16] S. D. Gollapalli, C. Caragea, Extracting keyphrases from research papers using citation networks, in: Proceedings of the AAAI conference on artificial intelligence, volume 28, 2014.

[17] S. N. Kim, O. Medelyan, M.-Y. Kan, T. Baldwin, SemEval-2010 Task 5 : Automatic Keyphrase Extraction from Scientific Articles, in: K. Erk, C. Strapparava (Eds.), Proceedings of the 5th International Workshop on Semantic Evaluation, Association for Computational Linguistics, Uppsala, Sweden, 2010, pp. 21–26. URL: https://aclanthology.org/S10-1004.

[18] I. Augenstein, M. Das, S. Riedel, L. Vikraman, A. Mc-

Callum, SemEval 2017 Task 10: ScienceIE - Extracting Keyphrases and Relations from Scientific Publications, in: S. Bethard, M. Carpuat, M. Apidianaki, S. M. Mohammad, D. Cer, D. Jurgens (Eds.), Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 546–555. URL: https://aclanthology.org/S17-2091. doi:10.18653/v1/S17-2091.

[19] A. Hulth, Improved Automatic Keyword Extraction Given More Linguistic Knowledge, in: Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing, 2003, pp. 216–223. URL: https://aclanthology.org/W03-1028.

[20] A. Oliver, M. Vàzquez, A free terminology extraction suite, in: Proceedings of Translating and the Computer 29, 2007.

[21] J. A. Lossio-Ventura, C. Jonquet, M. Roche, M. Teisseire, Combining c-value and keyword extraction methods for biomedical terms extraction, in: LBM: languages in biology and medicine, 2013.

[22] J. S. Justeson, S. M. Katz, Technical terminology: some linguistic properties and an algorithm for identification in text, Natural language engineering 1 (1995) 9–27.

[23] D. Bourigault, Surface grammatical analysis for the extraction of terminological noun phrases, in: COLING 1992 Volume 3: The 14th International Conference on Computational Linguistics, 1992.

[24] K. Kageura, E. Marshman, Terminology extraction and management, in: The Routledge Handbook of Translation and Technology, Routledge, 2019, pp. 61–77.

[25] B. Daille, Conceptual structuring through term variations, in: Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment, 2003, pp. 9–16.

[26] F. Sclano, P. Velardi, Termextractor: a web application to learn the shared terminology of emergent web communities, in: Enterprise Interoperability II, Springer, 2007, pp. 287–290.

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

[28] M. T. Pazienza, M. Pennacchiotti, F. M. Zanzotto, Terminology extraction: an analysis of linguistic and statistical approaches, in: Knowledge mining, Springer, 2005, pp. 255–279.

[29] L. P. Jones, E. W. Gassie, Jr, S. Radhakrishnan, Index: The statistical basis for an automatic conceptual phrase-indexing system, Journal of the American Society for Information Science 41 (1990) 87–97.

[30] A. Oliver, M. Vázquez, J. Moré, Linguoc lexterm: una herramienta de extracción automática de terminología gratuita, Translation Journal (2007).

[31] J. Rocheteau, B. Daille, Ttc termsuite: A uima application for multilingual terminology extraction from comparable corpora, in: 5th International Joint Conference on Natural Language Processing (IJCNLP), 2011, pp. 9–12.

[32] J. Vivaldi, H. Rodríguez, Improving term extraction by combining different techniques, Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication 7 (2001) 31–48.

[33] R. Mihalcea, P. Tarau, Textrank: Bringing order into text, in: Proceedings of the 2004 conference on empirical methods in natural language processing, 2004, pp. 404–411.

[34] R. Campos, V. Mangaravite, A. Pasquali, A. Jorge, C. Nunes, A. Jatowt, Yake! keyword extraction from single documents using multiple local features, Information Sciences 509 (2020) 257–289. doi:10.1016/j.ins.2019.09.013.

[35] H. H. Alrehamy, C. Walker, Semcluster: unsupervised automatic keyphrase extraction using affinity propagation, in: Advances in Computational Intelligence Systems: Contributions Presented at the 17th UK Workshop on Computational Intelligence, September 6-8, 2017, Cardiff, UK, Springer, 2018, pp. 222–235.

[36] A. Bougouin, F. Boudin, B. Daille, Topicrank: Graph-based topic ranking for keyphrase extraction, in: International joint conference on natural language processing (IJCNLP), 2013, pp. 543–551.

[37] C. Florescu, C. Caragea, A position-biased pagerank algorithm for keyphrase extraction, in: Proceedings of the AAAI conference on artificial intelligence, volume 31, 2017.

[38] B. Wang, S. Yu, W. Lou, Y. T. Hou, Privacy-preserving multi-keyword fuzzy search over encrypted data in the cloud, in: IEEE INFOCOM 2014-IEEE conference on computer communications, IEEE, 2014, pp. 2112–2120.

[39] D. Mahata, J. Kuriakose, R. Shah, R. Zimmermann, Key2vec: Automatic ranked keyphrase extraction from scientific articles using phrase embeddings, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), 2018, pp. 634–639.

[40] D. Gromann, L. Wachowiak, C. Lang, B. Heinisch, Multilingual extraction of terminological concept systems, Deep Learning and Neural Approaches for Linguistic Data (2021) 5.

[41] Y. Sun, H. Qiu, Y. Zheng, Z. Wang, C. Zhang, Sifrank: A new baseline for unsupervised keyphrase extraction based on pre-trained language model, IEEE Access 8 (2020) 10896–10906.

doi:10.1109/ACCESS.2020.2965087.
[42] C. Lang, L. Wachowiak, B. Heinisch, D. Gromann, Transforming term extraction: Transformer-based approaches to multilingual term extraction across domains, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, 2021, pp. 3607–3620.
[43] A. Ghafoor, A. S. Imran, S. M. Daudpota, Z. Kastrati, R. Batra, M. A. Wani, et al., The impact of translating resource-rich datasets to low-resource languages through multi-lingual text processing, IEEE Access 9 (2021) 124478–124490.
[44] L. Bonifacio, V. Jeronymo, H. Q. Abonizio, I. Campiotti, M. Fadaee, R. Lotufo, R. Nogueira, mmarco: A multilingual version of the ms marco passage ranking dataset, arXiv preprint arXiv:2108.13897 (2021).
[45] M. Araújo, A. Pereira, F. Benevenuto, A comparative study of machine translation for multilingual sentence-level sentiment analysis, Information Sciences 512 (2020) 1078–1102.
[46] C. P. Carrino, M. R. Costa-Jussà, J. A. Fonollosa, Automatic spanish translation of the squad dataset for multilingual question answering, arXiv preprint arXiv:1912.05200 (2019).
[47] G. M. Rosa, L. H. Bonifacio, L. R. de Souza, R. Lotufo, R. Nogueira, A cost-benefit analysis of cross-lingual transfer methods, arXiv preprint arXiv:2105.06813 (2021).
[48] 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).
[49] A. Gutiérrez-Fandiño, J. Armengol-Estapé, M. Pàmies, J. Llop-Palao, J. Silveira-Ocampo, C. P. Carrino, A. Gonzalez-Agirre, C. Armentano-Oller, C. Rodriguez-Penagos, M. Villegas, Maria: Spanish language models, arXiv preprint arXiv:2107.07253 (2021).
[50] C. Toraman, E. H. Yilmaz, F. Şahinuç, O. Ozcelik, Impact of tokenization on language models: An analysis for turkish, ACM Trans. Asian Low-Resour. Lang. Inf. Process. 22 (2023). URL: https://doi.org/10.1145/3578707. doi:10.1145/3578707.

A. Term Translation Prompt

You are a scientific translator of English to Spanish specialized in terminology. I give you one sentence in English and the same sentence translated to Spanish. The English sentence has a term between the marks
 and </br>. Identify in the Spanish sentence which words correspond to the same original term. The output term is in Spanish. Some examples

English sentence: "The University of Florida, in partnership with Motorola, has held two
mobile computing</br> design competitions". Spanish sentence : "La Universidad de Florida, en asociación con Motorola, ha celebrado dos concursos de diseño de computación móvil". Output: computación móvil English sentence: "There, we assume that
coefficients of non-renormalizable terms</br> are suppressed enough to be neglected". Spanish sentence: "Aquí, asumimos que los coeficientes de los términos no renormalizables están suficientemente suprimidos como para ser ignorados". Output: coeficientes de los términos no renormalizables

English sentence: "It often exploits an
optical diffusion model-based image reconstruction algorithm</br> to estimate spatial property values from measurements of the light flux at the surface of the tissue." Spanish sentence: "A menudo se utiliza un algoritmo de reconstrucción de imágenes basado en un modelo de difusión óptica para estimar los valores de propiedades espaciales a partir de medidas de la flujo de luz en la superficie del tejido." Output: algoritmo de reconstrucción de imágenes basado en un modelo de difusión óptica

English: "A second group of experiments is aimed at extensions of the baseline methods that exploit characteristic features of the UvT Expert Collection; specifically, we propose and evaluate refined expert finding and profiling methods that incorporate
topicality and organizational structure</br>." Spanish: "Un segundo grupo de experimentos está dirigido a extensiones de los métodos base que aprovechan las características distintivas de la Colección de Expertos de UvT; específicamente, proponemos y evaluamos métodos refinados de búsqueda y perfilado de expertos que incorporan la topicalidad y la estructura organizativa." output: topicalidad y la estructura organizativa