Contents lists available at ScienceDirect

Artificial Intelligence In Medicine

journal homepage: www.elsevier.com/locate/artmed

A keyphrase-based approach for interpretable ICD-10 code classification of Spanish medical reports

Andres Duque [a,b,*], Hermenegildo Fabregat [a], Lourdes Araujo [a,b], Juan Martinez-Romo [a,b]

[a] Universidad Nacional de Educación a Distancia (UNED). ETS Ingeniería Informática, Juan del Rosal 16, 28040 Madrid, Spain
[b] Instituto Mixto de Investigación - Escuela Nacional de Sanidad (IMIENS), Spain

ARTICLE INFO

Keywords:
Medical records
ICD-10 codes
Keyphrase extraction
Interpretability

ABSTRACT

Background and objectives: The 10th version of International Classification of Diseases (ICD-10) codification system has been widely adopted by the health systems of many countries, including Spain. However, manual code assignment of Electronic Health Records (EHR) is a complex and time-consuming task that requires a great amount of specialised human resources. Therefore, several machine learning approaches are being proposed to assist in the assignment task. In this work we present an alternative system for automatically recommending ICD-10 codes to be assigned to EHRs.

Methods: Our proposal is based on characterising ICD-10 codes by a set of keyphrases that represent them. These keyphrases do not only include those that have literally appeared in some EHR with the considered ICD-10 codes assigned, but also others that have been obtained by a statistical process able to capture expressions that have led the annotators to assign the code.

Results: The result is an information model that allows to efficiently recommend codes to a new EHR based on their textual content. We explore an approach that proves to be competitive with other state-of-the-art approaches and can be combined with them to optimise results.

Conclusions: In addition to its effectiveness, the recommendations of this method are easily interpretable since the phrases in an EHR leading to recommend an ICD-10 code are known. Moreover, the keyphrases associated with each ICD-10 code can be a valuable additional source of information for other approaches, such as machine learning techniques.

1. Introduction

The International Classification of Diseases (ICD), developed by the World Health Organization (WHO), is a list of medical classification codes for diseases, signs and symptoms, abnormal findings, complaints, social circumstances, and external causes of injury or diseases. The codes are organised in two categories, diagnoses and procedures. The ICD codes have been widely adopted by doctors and other health professionals for reimbursement, storage, and retrieval of diagnostic information. ICD-10[1] is the 10th revision of the ICD, and the Spanish version of this revision is called CIE-10-ES. It consists of a hierarchical alphanumeric classification containing between 3 and 7 digits, collecting more detailed clinical information than previous versions. The number of diagnoses considered is 69,099, and the number of procedures is 72,000. This large variability implies great complexity in the selection of codes for the Electronic Health Records (EHRs) corresponding to a patient's visit.

The process of manually assigning ICD codes to a EHR is complex and time-consuming. Expert annotators need to look for key information in the EHRs which can be a long, unstructured text. Afterwards, they have to select the codes to be assigned from a huge hierarchy, taking into account aspects such as the anatomic site, the severity, and the etiology [1]. Hence, the development of automatic systems to help this coding process has become an important necessity within the field. These systems require high reliability and accuracy in selecting the codes. A good coverage is also desirable, so that all relevant codes are collected, from which coding experts can select the most suitable ones. Due to the huge number of possible codes that can be assigned to a particular EHR, this

* Corresponding author at: Universidad Nacional de Educación a Distancia (UNED), ETS Ingeniería Informática, Juan del Rosal 16, 28040 Madrid, Spain.
E-mail addresses: aduque@lsi.uned.es (A. Duque), gildo.fabregat@lsi.uned.es (H. Fabregat), lurdes@lsi.uned.es (L. Araujo), juaner@lsi.uned.es (J. Martinez-Romo).

[1] https://www.who.int/classifications/icd/factsheet/en/

https://doi.org/10.1016/j.artmed.2021.102177
Received 21 September 2020; Received in revised form 14 September 2021; Accepted 14 September 2021
Available online 22 September 2021

process can be seen as an Extreme Multi-label Classification task (XMLC). This is a particularly difficult case of text classification in which a system must find the most relevant subset of classes to which a particular document (in this case, the EHR) belongs, among an extremely large space of categories (from thousands to millions of them) [2].

Many of the proposals developed are based on supervised machine learning systems [3], including the more recent deep learning systems [4]. However, in the medical field, the predictions of automatic systems need to be explained [5], so that doctors can trust them and explain the decisions made. In this context, it is important to provide evidence of the contents of an EHR that have led to the selection of each code assigned to it.

With this goal in view, we have designed an ICD-10 code recommendation system for the Spanish language based on the identification of keyphrases that characterise the EHR. Keyphrases are phrases that represent the content of a document. They may be composed of more than one word.

The idea of this proposal is to accurately collect all the keyphrases that characterise an ICD-10 code. More specifically, we aim to generate a set of keyphrases associated to each ICD-10 code, so that when a new report needs to be processed, those codes significantly related to the keyphrases extracted from the new report are selected.

Some aspects that difficult this task are the following:

- The keyphrases extracted may take different lexical forms (for instance, they may present different endings), even if they refer to the same entity or concept. In a similar way, equivalent concepts may be expressed using different word orders. It is therefore necessary to identify variant forms of the same concept for calculating its degree of relatedness with each ICD-10 code.
- As mentioned before, the total number of classes (in this case, ICD-10 codes) is very high, and moreover many different codes can be assigned to a particular EHR. It is crucial to identify those keyphrases detected in a report that unequivocally correspond to each ICD-10 assigned to it, as well as to differentiate them from those keyphrases that may correspond to many different codes and hence are not useful in the classification process.
- Medical reports are usually presented as unstructured or poorly structured pieces of text, containing both important information and additional data which can be omitted to avoid introducing noisy information into the classification process. Hence, techniques that identify those parts of the reports containing valuable information should be explored.

The system we propose involves the identification of keyphrases in EHRs. We apply an algorithm based on statistical techniques to identify the association of ICD-10 codes and keyphrases that have a high statistical significance. Finally, associations between ICD-10 codes and representative keyphrases are made based on this significance, in order to assign the correct ICD-10 codes to new reports. Although the total number of possible ICD-10 codes is very high (around 141,000, as mentioned before), in the particular problem addressed in this work the number of considered codes will be around 8,000, as it will be explained in Section 4.1. All the reports considered in this work are discharge reports from different hospital services, such as internal medicine, obstetrics and gynecology, gastroenterology, cardiology and many others. Hence, the codes to be identified in the task also correspond to those different hospital services.

The main contributions of this work are the following:

- We present a complete pipeline for classifying medical reports written in the Spanish language, achieving state-of-the-art accuracy.
- A method of statistical significance that allows establishing weighted relationships between ICD-10 codes and keyphrases is applied to the addressed problem with promising results.
- The high interpretability of the results provided by our system makes it a particularly useful and reliable tool for the medical domain.
- We model the ICD-10 code classification problem as an extreme multilabel classification task, addressing its data imbalance issues and proposing different test frameworks and metrics for minimising their impact.

Our approach to the problem of assigning ICD-10 codes to medical reports is based on Natural Language Processing techniques and statistical analysis algorithms characteristic of Artificial Intelligence. At a time when most of the work devoted to the classification of documents follows deep learning techniques, our proposal provides an alternative path with competitive results. This new approach can not only be combined with other techniques, but also has the great advantage of providing an intuitive justification of the codes selected for a report, which health professionals can easily analyse. This feature is essential in the health domain, as predictions provided by automatic systems must be accompanied by information that justifies them.

The rest of the paper is structured as follows: Section 2 presents an overview of previous works that can be found in the literature facing similar problems. The whole system developed in this work, broken down into all its component modules, is described in detail in Section 3. Section 4 is devoted to introducing the dataset and metrics considered for evaluating our system, as well as to showing the achieved results and their comparison with similar systems. Finally, Section 5 presents some conclusions and future lines of research.

2. Background and previous work

Automatic ICD coding has been addressed in many recent works. Xu et al. [6] distinguished different modalities of source data depending on the degree of structure: unstructured text, semi-structured text and structured tabular data. They developed separate machine learning models for each modality. Then, they applied an ensemble method to integrate all modality-specific models to generate ICD-10 codes. The dataset used in that work was MIMIC-III [7], containing information from about 58,000 hospital admissions reports of patients staying in the ICU of the Beth Israel Deaconess Medical Center between 2001 and 2012. They tried to provide clues for the interpretability of the results by computing weights for the links between words in the reports and the ICD codes. This weight is calculated in function of the values of the hidden units from a trained neural network. These weights are calculated based on the values of the neural network hidden units in the training phase. Other proposals [8] have taken advantage of the annotation with other coding systems, such as SNOMED-CT [9], for assigning ICD-10 codes. They evaluated the system using 5 years of EHRs obtained from three Australian hospitals, considering only the principal diagnosis codes, and up to a 5-character level of their ICD-10 codification. Two cross-maps were used to translate the clinical concepts annotated with MetaMap [10] and NegEx [11] into ICD-10-AM codes. They correspond to NLM's UMLS [12] to ICD-10-AM mapping tables and to an in-house version of SNOMED CT to ICD-10-AM map. Subotin and Davis [13] predicted ICD-10 procedure codes by supplementing sparse ICD-10 training resources with ICD-9 data. They applied a partial hierarchical classification to identify potentially relevant concepts and codes. Then, confidence values for the candidate codes were estimated by means of a model trained on data with ICD-9 codes.

CLEF[2] has organised some shared tasks devoted to a large scale classification task consisting of extracting causes of death as coded in ICD-10 [14,15,16]. Task 2 of the 2016 CLEF eHealth evaluation lab [14] introduced a large-scale classification task in French death certificates. The data provided to the participants consisted of a few lines of text with at least one main diagnosis. Multiple different codes could be assigned to

[2] http://www.clef-initiative.eu/

a particular line. The goal of the task consisted of mapping the sentences containing one or more diagnoses to their corresponding ICD-10 codes. All five participant teams were evaluated using 27,850 death certificates and measures of Precision, Recall and F-measure. The highest performance reached in this edition was 0.848 F-measure. In the 2017 edition, the task [15] focused on English and French. Eleven teams participated, 10 of them submitted runs for the English dataset and 9 for the French dataset. The participants proposed different methods. Some of them used lexical resources such as the dictionaries supplied as part of the training data as well as other medical terminologies and ontologies, while other teams adopted a classical supervised approach, exploiting only the gold standard training data for training machine learning systems, some of them using deep learning techniques. The highest performance reached for French was 0.8674, obtained by team LIMSI using a machine learning method relying on knowledge based-sources. The highest performance for English, 0.8501, was reached by the KFU team using Recurrent Neural Networks. In 2018 [16], the task was focused on French, Hungarian, and Italian. Most participants relied on machine learning methods such as neural networks. Other techniques considered were information retrieval and dictionary and ontology matching. Most participants made use of the dictionaries supplied along with the training data. IxaMed team [4] obtained the best performance in terms of F-measure for all datasets: 0.838 for French, 0.9627 for Hungarian and 0.9524 for Italian. They applied a neural model for mapping the input text snippets with the output ICD-10 codes. Only three participant teams proposed to take advantage of the multilingualism in some manner. Two of them, WBI [17] and TorontoCL [18] used multilingual word embeddings for dealing with multiple languages simultaneously. Another team, KCL [19] employed French data to pre-train an encoder-decoder architecture for Italian death certificates. In additional out of competition proposals using the CLEF 2018 shared task data, Almagro et al. [20] deployed a transfer learning approach using Machine Translation for similar documents. Specifically, the system uses death certificates from one language to code certificates in another language. In this way the amount of available coded documents is increased, improving the system performance. Atutxa et al. [21] tackled the problem as a sequence-to-sequence task, testing different types of neural architectures for the three CLEF 2018-Task 1 datasets. They outperformed the best results of the participating teams, providing at the same time clues for interpreting the results given by the alignments between the original text and each output code. A new task for assigning ICD-10 codes to medical reports written in Spanish has also been proposed in the 2020 CLEF conference [22]. Best performing systems use machine learning techniques such as XGBoost combined with string similarity techniques [23], or even more classical NLP pipelines based on the use of dictionaries especially built for the task [24], as well as deep learning methods based on pre-trained multilingual language models [25].

Other works, not purely based on classifiers, focus on requiring lesser degree of supervision. For example, Ning et al. [26] took advantage of the hierarchical structure of ICD-10 codes and proposed a hierarchy-based method that determines the most suitable code until obtaining the subcategory code. The code assignment is based on a measure of semantic similarity computed using HowNet, a Chinese domain-independent knowledge base [27]. Similarly, Chen et al. [28] also proposed an approach based on semantic similarity. They used the Longest Common Subsequence (LCS) to compute semantic similarity for automatic Chinese diagnoses, mapping from the disease names given by clinicians to their corresponding ICD-10 codes. LCS is the longest string that is a subsequence of every member of a given set of strings. Almagro et al. [29] propose an approach based on information retrieval, by indexing the texts of the ICD-10 code descriptions. They use a dataset very similar to the one presented in this work which, as it will be described later, is a much more complex dataset than those used in the abovementioned shared tasks. The experiments undertaken indicate that such an approach is able to treat cases where the frequency of occurrence of ICD-10 codes is not as high as required by the classification systems.

Automatic keyphrase (also known as key terms and keywords) extraction aims to identify a set of phrases that capture the main topics of a document or a set of documents [30,31]. Keyphrases provide a concise summary of the document content and thus allow to get its main ideas. They have many important applications, including searching, document clustering and classification. Since the manual extraction of keyphrases is an expensive and time-consuming task, there have been many proposals to automate the process. Unsupervised techniques have been frequently applied for this task, such as statistical-based TF-IDF [32,33], and graph-based techniques [34,35,36]. In recent years, word embedding and neural networks have also been applied to address the problem [37,38]. Some works have tackled the problem specifically in the biomedical domain [39]. In this work we propose using and combining different approaches for then applying them, not to scientific articles, as usual, but to medical reports. As we will see, the different approaches used complement and improve the individual results of each of them.

Finally, the use of rules, and more particularly rules based on manually curated regular expressions, for detecting sections within text is a widely used technique in the biomedical research in NLP. A detailed survey presented by [40] offers precise information regarding different techniques used for section identification of clinical reports. Out of a total of 39 studies on the subject, 59% of them were rule based systems, and from them, almost 3 out every 4 used regular expressions. These expressions are usually devoted to locate headings through the use of heuristics based on the analysis of casing information, punctuation marks such as semicolons or hyphens, line breaks, and so on. Rules are manually created for detecting the start and end of particular sections in [41]. These rules are designed after an analysis of a representative sample of the collection. Another example of systems using regular expressions for detecting sections can be seen in [42], in which the section of interest, named “Impression”, is extracted from each report by applying hand-crafted rules. Rule-based methods are normally validated using a sample of the available texts, before applying these rules to the whole dataset. For instance, in the work by [43], a subset of 200 documents from a total of 5271 is randomly selected for analysis and generation of manual regular expressions. As it will be explained later on, a similar methodology for analysing the improvements offered by an initial segmentation of the considered medical reports has been followed in this research.

3. System description

In this section the pipeline used for assigning ICD-10 codes to medical reports is depicted, and the different submodules developed for each phase of the pipeline are described. Fig. 1 shows the general diagram representing the whole process. A subset of labelled reports (this is, reports for which the associated ICD-10 codes are known) is considered for building the knowledge base that will eventually be used for classification (top part of the figure). In the pre-processing step, different sections of those reports are extracted in order to discriminate sections containing the most valuable medical information from those containing uninformative aspects of the clinical case (segmentation process). Then, various techniques are applied for extracting the most significant keyphrases of each medical report. These keyphrases are subsequently used in the representation step for obtaining a knowledge base in which keyphrases are assigned to the different ICD-10 codes considered, in accordance with their statistical significance, as mentioned in Section 1. This statistical significance is transformed into a weight representing the strength of the relationship between a keyphrase and an ICD-10 code. Once this representation framework is built, some additional information is added to the knowledge base in a post-processing step for refining it, considering different variations of the extracted keyphrases, according to different lexical and semantic forms.

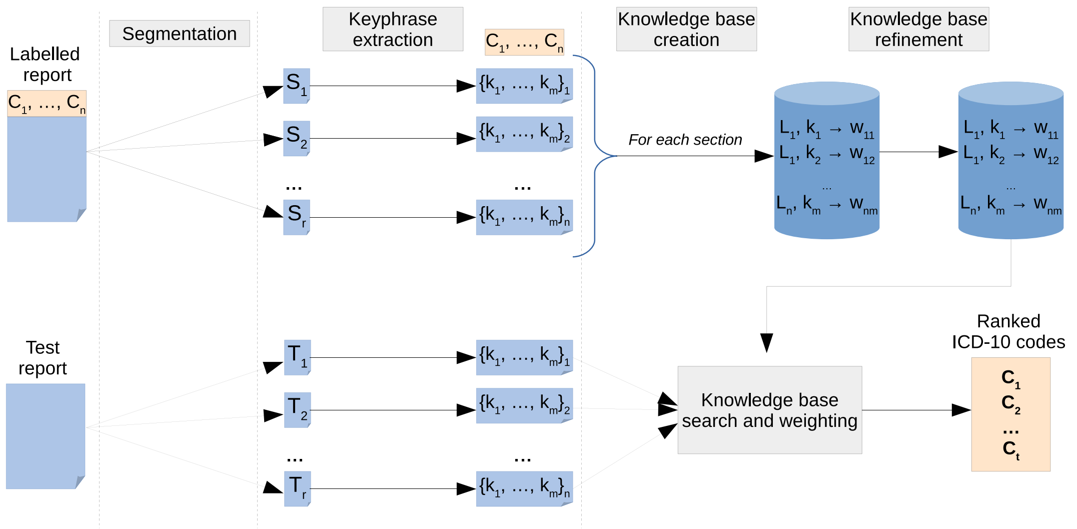


Fig. 1. Flow diagram of the process for recommending ICD-10 codes to medical reports. The knowledge base creation is shown in the top part of the figure, and the classification of new test reports can be seen in the bottom part of the figure.

The last step (bottom part of Fig. 1) of the pipeline is the classification of new reports that have not been previously seen by the system. Keyphrases are extracted from the new report for characterising it, and candidate ICD-10 codes are selected from the knowledge base and assigned to the report, according to their statistical relationships with the extracted keyphrases. The ICD-10 codes are weighted using the information in the knowledge base and finally ranked by their total weight.

3.1. *Pre-processing*

The pre-processing step involves several Natural Language Processing (NLP) techniques applied to the textual information in the medical reports for extracting the most valuable pieces of medical data related to each of the possible ICD-10 codes used in the classification process.

3.1.1. *Report segmentation*

An initial segmentation step is applied over the raw text of the reports for separating them into different sections. Those sections will eventually represent diverse parts of the information registered by health professionals when assessing the follow-up of a particular patient: personal and family information, main and secondary reasons for medical consultation and admission, clinical judgment or treatment, among others. Since the considered medical reports do not follow a clear or particularly homogeneous structure, the automatic identification of appropriate sections within those reports is difficult to achieve. However, we have performed a previous study on a subset of the reports used for building the knowledge base, in order to identify some commonly used sections, as well as some interesting textual expressions usually related to the beginning or end of those sections. In this study, a total of 200 medical reports randomly extracted from the training dataset have been used for generating the regular expressions used in this step of the research. More specifically, 150 reports have been used for manually developing regular expressions, taking into account how the information is distributed along the reports and the way the information is usually presented in them. The remaining 50 reports have been kept for manually analysing the performance of those proposed regular expressions on unseen reports. Hence, the first 150 reports can be seen as the training dataset for this step, and the remaining reports to be the test dataset. From this information, a splitting process based on the use of regular expressions has been implemented for automatically detecting these main sections.

The developed regular expressions consist of two parts: format modifiers including hyphens or tabs, and literals defining those headers that usually appear in the reports when a section is presented. For instance, section “Treatment” is identified through the expression “*Tratamiento habitual*”, which usually appears at the beginning of a sentence, and is followed by symbols, blank spaces or line separators. A similar regular expression used for identifying the same section would be built with the expression “*Medicación actual*” (“current medication”). In a similar way, the appearance of the sentence “*Factor de riesgo cardiovascular*”, or its abbreviation “*FRCV*” (which stands for “cardiovascular risk factor”) usually indicates the beginning of a section devoted to describe the patient's risk of developing cardiovascular diseases due to particular factors. The section that describes reasons for medical consultation and admission can be detected by finding the sentence “*Motivo de consulta*” (literally, “reason for consultation”), but also by finding the phrase “*ingresada para*” (which stands for “admitted for”).

For each sentence within a report, the segmentation utility extracts the different possible sections that match with one or more regular expressions found in the text. The beginning of a section corresponds to the appearance of a particular expression, and its ending with the appearance of another expression associated with a different section. This way, discontinuous parts of the same section can also be found and merged in a final step.

A total of 12 sections have been identified through this process. Some examples of them are “*Antecedentes e historia clínica personal/familiar*” (“Background and personal/family clinical history”), or “*Juicio clínico*” (“Clinical judgment”). A section named “Unidentified section” includes all the information from the medical report that has not been identified by the segmentation utility. Despite being automatically detected in almost all the considered reports, it usually contains non-informative data such as anonymised information (represented with the string “*xxxx*”) or isolated sentences.

The complete list of sections extracted with the segmentation utility can be seen in Appendix A.

3.1.2. *Keyphrase extraction*

Two different methods have been explored and combined for the extraction of representative keyphrases of each medical report. The first method is based on selecting the most relevant words and phrases according to the TF-IDF technique [44]. The main advantage of this approach lies on its ability to adapt to the considered document collection, and therefore to its particular vocabulary. Some standard pre-processing steps are applied prior to the keyphrase extraction step, such as transforming the words in the text to lemmas, lowercasing them and eliminating symbols and accents. Through the lemmatisation step, we are able to reduce different verb tenses to their corresponding infinitives, convert plural forms of the words to their singular forms, or eliminate non-informative prefixes and suffixes from particular nouns. Apart from it, a Part-Of-Speech (POS) tagging of the text will also be needed in order to begin the keyphrase identification process. Both lemmatization and POS-tagging of the original text are carried out at the same time using the TreeTagger tool [45].

A regular expression for capturing useful keyphrases related to medical concepts and procedures is defined. This expression identifies a subset of grammatically correct Spanish noun and prepositional phrases, which correspond to the usual forms in which these concepts are found within medical texts. The specific form of the regular expression is the following:

$$(\text{NEG? JJ* } (\text{NN.*})^{+} \text{ JJ* IN})\text{? JJ* } (\text{NN.*})^{+} \text{ JJ*}$$

where "NEG" represents any negation trigger such as "*no*", "*ni*" or "*sin*", "JJ" represents an adjective, "NN" represents a noun and "IN" indicates the use of a preposition. This way, when the first part (up to the preposition) appears in the sentence a prepositional phrase will be found (for instance, "*dificultad para respirar*", literally "difficulty in breathing"), while the non-appearance of this first part will indicate a single noun or a noun phrase. The process of detecting keyphrases involves the use of a chunker for parsing the sentences and detecting sequences of POS tags which match the regular expression. The tool used for this process is the python library NLTK [46], which includes a regular expression parser and allows to convert any sequence of POS tags into the "CoNLL" IOB tag format,[3] which is subsequently used for building the final candidate keyphrases. Hence, this process would be akin of a parsing which uses a custom regular expression for avoiding the detection of keyphrases which would not be of interest for the purposes of the system.

In the final step, a TF-IDF model is built on the candidate keyphrases generated from the labelled reports for selecting the most representative keyphrases, associated in turn with their respective report sections.

The second keyphrase extraction method makes use of the tool IXAMedTagger,[4] presented in [47] and used in subsequent works such as [48,49,50]. IXAMedTagger is a medical tagger based on the open source multilingual NLP library FreeLing, adapted for the Spanish language [51] and the biomedical domain [52]. This tool can be used for automatically annotating biomedical concepts from the Spanish version of the Systematized Nomenclature of Medicine Clinical Terms (SNOMED-CT). It is based on the Perceptron algorithm [53,54] and provides information regarding 4 types of different medical entities: drugs and substances, diseases and symptoms, anatomical elements associated with diseases, and qualifiers also associated with diseases. The use of this method allows us to complete the set of keyphrases initially obtained with the TF-IDF model based on regular expressions, with a more specialised subset of candidate phrases provided with the Perceptron-based tool.

Fig. 2 shows an example of the different keyphrases extracted from an excerpt of a particular medical report, using both aforementioned keyphrase extraction methods: TF-IDF and IXAMedTagger.

Different and somehow complementary keyphrases are extracted using both methods: the TF-IDF model allows us to extract general keyphrases according to the different writing styles presented by health professionals when filling out a medical report. However, it sometimes misses medical concepts such as drug names (in the example, "Fergradumet"). The Perceptron-based tagger, on the other hand, offers more objective keywords and keyphrases extracted from medical knowledge bases, but is not able to capture more general keyphrases possibly helpful for further classification, such as "*curas de herida*" ("wound care practices"). As it will be shown in later sections, combining both methods generates an optimal set of keyphrases for performing ICD-10 code classification.

3.2. Knowledge base creation

Once all the keyphrases have been extracted from the labelled reports, a statistical process is applied for determining the most important keyphrases associated to each of the ICD-10 codes considered in the problem. The main framework used for this objective is described in [55], and has been successfully employed for extracting statistically based structured information in different scenarios. This statistical information can be used to infer semantic relationships between words in general purpose tasks [35,56] and also between medical concepts in the biomedical domain [57]. However, and unlike those previous works, in this case we intend to extract relationships between elements of different nature, particularly keyphrases and ICD-10 codes.

Hence, in this work, the main objective of this technique is the extraction of statistically significant indicators between the ICD-10 codes assigned to specific medical reports, divided into sections following the procedure described in Section 3.1.1, and the keyphrases of those reports, extracted as explained in Section 3.1.2. Thus, we will consider a document to be the set of keyphrases from a section of a medical report, and the ICD-10 codes associated to that report. As we do not have any information regarding which codes should be assigned to each section of the report, the whole set of codes from the initial report is assigned to each section automatically extracted from it. Therefore, the statistical process will be applied for each of the possible sections previously defined, in order to generate a knowledge base linking ICD-10 codes and their most representative keyphrases, for each of the considered sections.

Each document thus can be seen as a "bag of concepts" containing two types of elements, namely keyphrases (appearing in the report) and ICD-10 codes (assigned to the report). The method that is proposed in this work is based on considering the co-occurrence of pairs of elements within the documents, and extracting the statistical significance of this co-occurrence depending on the individual occurrences of each element separately. By forcing each pair of elements to consist of a keyword and an ICD-10 code, we expect to accurately model how important will be the relationship between those elements appearing together in a particular number of documents, in relation to their separate appearance in other documents.

Formally, we consider two elements e_1 and e_2, of which we know that one of them is a keyword and the other one is an ICD-10 code, to be appearing in n_1 and n_2 documents, respectively, and co-occurring in k documents. Hence, four different types of documents will be considered: $n_1{-}k$ documents showing only element e_1, $n_2{-}k$ documents showing only e_2, k documents presenting a co-occurrence of both elements, and $N - n_1 - n_2 + k$ documents with none of them, given that N is the total number of documents considered for building the knowledge base.

The number of possible combinations is given by the following multinomial coefficient:

$$\begin{pmatrix} N \\ k, n_1 - k, n_2 - k \end{pmatrix} = \begin{pmatrix} N \\ k \end{pmatrix} \begin{pmatrix} N - k \\ n_1 - k \end{pmatrix} \begin{pmatrix} N - n_1 \\ n_2 - k \end{pmatrix} \tag{1}$$

And the probability of two elements co-occurring exactly k times by pure chance is:

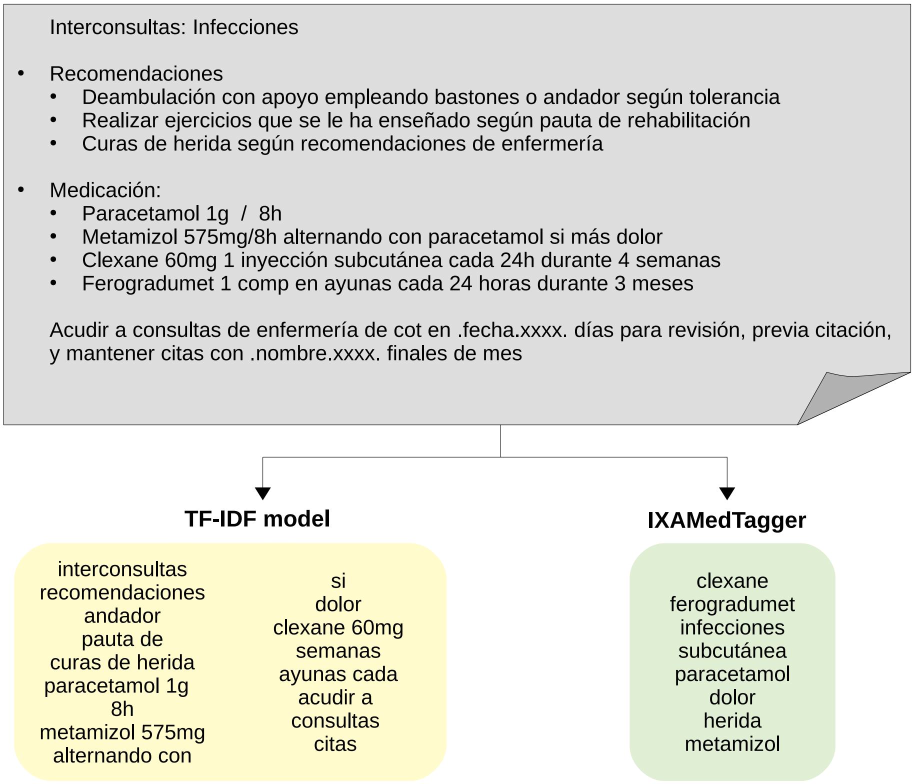
$$p\left(k\right) = \begin{pmatrix} N \\ n_1 \end{pmatrix}^{-1} \begin{pmatrix} N \\ n_2 \end{pmatrix}^{-1} \begin{pmatrix} N \\ k, n_1 - k, n_2 - k \end{pmatrix} \tag{2}$$

if $max(0, n_1 + n_2 - N) \leq k \leq min\ (n_1, n_2)$ and zero otherwise.

Given these equations (Eqs. (1) and (2)), and assuming $n_1 \geq n_2 \geq k$ (without any loss of generality), this probability can be computed as follows:

[3] https://www.nltk.org/api/nltk.chunk.html

[4] http://ixa2.si.ehu.eus/prosamed/resources



Interconsultas: Infecciones

- Recomendaciones
  - Deambulación con apoyo empleando bastones o andador según tolerancia
  - Realizar ejercicios que se le ha enseñado según pauta de rehabilitación
  - Curas de herida según recomendaciones de enfermería
- Medicación:
  - Paracetamol 1g / 8h
  - Metamizol 575mg/8h alternando con paracetamol si más dolor
  - Clexane 60mg 1 inyección subcutánea cada 24h durante 4 semanas
  - Ferogradumet 1 comp en ayunas cada 24 horas durante 3 meses

Acudir a consultas de enfermería de cot en .fecha.xxxx. días para revisión, previa citación, y mantener citas con .nombre.xxxx. finales de mes

Fig. 2. Keyphrases extracted from a medical report using the two explored methods: TF-IDF model and IXAMedTagger.

$$p\left(k\right) = \prod_{j=0}^{n_2-k-1}\left(1-\frac{n_1}{N-j}\right)\times\prod_{j=0}^{k-1}\frac{(n_1-j)(n_2-j)}{(N-n_2+k-j)(k-j)} \tag{3}$$

Hence, Eq. (3) shows the probability of the pair of elements co-occurring exactly k times by pure chance, which can be defined as our null model. Then, the p-value for the co-occurrence of two elements e_1 and e_2 can be defined as follows:

$$p=\sum_{k\geq r}p\left(k\right) \tag{4}$$

where r is the actual number of co-occurrences found in our corpus between e_1 and e_2. From the definition of our null model, we will be able to refute it when the p-value lies below a defined threshold p_0 next to 0. This will indicate that the relationship between the two considered elements e_1 and e_2 (a keyword and an ICD-10 code) is statistically significant and hence should be taken into account and included in the knowledge base. A threshold of p_0=0.01 has been considered for all of the experiments conducted in this work, which means that the confidence of each relationship included in the knowledge base is over 99%.

This computation of the p-value is equivalent to calculate the survival function of a hypergeometric distribution, which is particularly appropriate for those cases in which the number of co-occurrences is small and element frequencies cannot be assumed to be normally distributed. Other co-occurrence based methods for calculating statistical significance such as Chi-Squared assume data to follow a Gaussian distribution, to which our data would only approximate for very large values.

By following this procedure, we are able to calculate all the p-values for pairs (e_1,e_2) in which e_1 is a ICD-10 code and e_2 a particular keyword, and hence we can sort by p-value all the keywords that are related to each ICD-10 code in a statistically significant manner. Fig. 3 shows keywords related to different ICD-10 codes within the knowledge base built for section "*Juicio clínico*" ("clinical judgment"). The description of each ICD-10 code is also included in the example in order to show its similarities and differences with the most important keywords associated to each code (those keywords whose associated p-values are smaller).

The precision of the co-occurrence method is illustrated by comparing the most relevant keyphrases associated to each ICD-10 code,

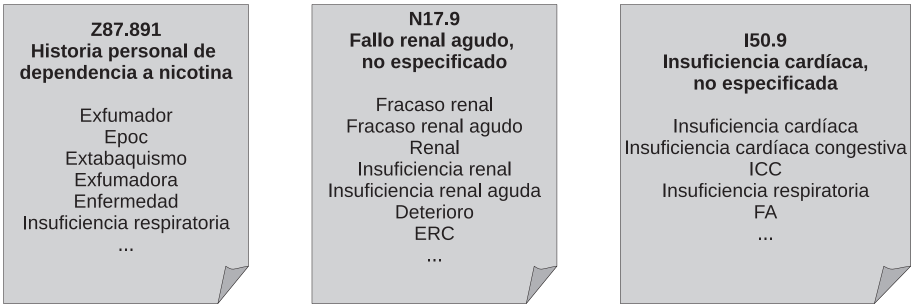

Z87.891
Historia personal de dependencia a nicotina
Exfumador
Epoc
Extabaquismo
Exfumadora
Enfermedad
Insuficiencia respiratoria
...
N17.9
Fallo renal agudo, no especificado
Fracaso renal
Fracaso renal agudo
Renal
Insuficiencia renal
Insuficiencia renal aguda
Deterioro
ERC
...
I50.9
Insuficiencia cardíaca, no especificada
Insuficiencia cardíaca
Insuficiencia cardíaca congestiva
ICC
Insuficiencia respiratoria
FA
...

Fig. 3. Most relevant (statistically significant) keyphrases associated to particular ICD-10 codes, for the knowledge base extracted from the "Clinical judgment" section. The description is shown below each ICD-10 code.

and the description of the code itself. In the first case, the code description is “Personal history of nicotine dependence”, and the keyphrases are “ex-smoker”, “EPOC” (Spanish acronym for “Chronic Obstructive Pulmonary Disease”), or “respiratory failure”. In the second example, the code description is “Acute kidney failure, unspecified”, and very similar keyphrases can be found: “renal failure”, “acute renal failure” or “ERC” (Spanish acronym for “Chronic Kidney Disease”). Finally, the code with description “Cardiac insufficiency, unspecified” is found to be associated with keyphrases such as “cardiac insufficiency”, “congestive cardiac insufficiency” and its acronym, “ICC”, or “FA” (Spanish acronym for “atrial fibrillation”).

3.3. *Post-processing*

The last step in the proposed pipeline is composed of a series of refinements performed on the knowledge bases created during the previous phases. These post-processing operations are mainly aimed at improving the coverage offered by the knowledge bases, this is, recognising as many correct keyphrases and concepts as possible once a new medical report is received by the system for classification. Therefore, the main objective of this phase is to enable the system to successfully use new information that was not available in the medical reports used for building the knowledge bases.

The actions taken in this post-processing step are related to the nature of the knowledge base itself, which is composed of pairs (ICD-10 code, keyphrase), with a particular *p*-value assigned to each pair. Considering that the same medical concept can be expressed using many different keyphrases and expressions, a grouping technique for gathering together all those expressions can be applied in order to reduce the need of finding an exact matching of a keyphrase in a test report for recognising it. To this end, different techniques are applied to the keyphrases that already form the knowledge base: stopword and non-informative keyphrase removal and bag-of-word transformation.

Apart from the classic stopword removal technique based on a list of stopwords related to a specific language, a more detailed analysis has been conducted on the medical reports used for building the knowledge bases in order to detect stopwords and non-informative expressions particularly related to the domain. For instance, keyphrases such as “*datos personales*” (“personal data”), “*centro sanitario*” (“healthcare facility”) or “*informe de medicina interna*” (“internal medicine report”) have been found to appear very frequently in the reports handled during this research, but are not informative enough for assigning any particular ICD-10 code to a report. Hence, we have performed a manual analysis of the most frequent keyphrases and removed those considered to be non-informative ones. A more exhaustive list of non-informative keyphrases removed in this step is shown in Appendix B.

Finally, the “bag-of-words transformation” technique basically transforms every keyphrase in the knowledge base to an equivalent bag of words in which the order of the words is not representative. The main reason for applying this technique is to address the high variability in the order of words in a phrase that is usually found in the Spanish language. For example, the same concept “severe headache” could be expressed using different keyphrases such as “*dolor de cabeza severo*”, “*severo dolor de cabeza*” or “*dolor severo de cabeza*”. By considering the keyphrase as a set of words {“*dolor*”}, {“*cabeza*”}, {“*severo*”} (preposition “*de*” would be removed as it is seen as a stopword), we are able to join together all those possible expressions into the same keyphrase, which will eventually lead to the detection of many of the possible ways to formulate a particular concept. Although this heuristic could lead to incorrectly consider as the same concept two keyphrases in which the word order is actually important for expressing different ideas, the conducted experiments have shown that the benefits of using this technique far outweigh its disadvantages.

After applying all these techniques, many sets of keyphrases, previously considered to be different, are now merged into the same keyphrase. Hence, a way to determine the new *p*-value assigned to the pair (ICD-10 code, keyphrase′) is needed, in which *keyphrase*′ represents the group of old keyphrases joined together. For this purpose, the minimum p-value of a pair (ICD-10 code, keyphrase) in which the keyphrase belongs to the initial set of keyphrases (before merging) will be selected to be the new p-value of the pair (ICD-10 code, keyphrase′).

It is important to assess the impact that this post-processing steps have on the final results, in order to illustrate its relevance within the whole system. To this end, some experiments will be shown in Section 4.3 both excluding and including these post-processing steps in the pipeline of the system, and the obtained results will be discussed.

3.4. *Report classification*

The pre-processing and post-processing steps are applied to each medical report in the test dataset for assigning the ICD-10 codes found by the system. The segmentation utility described in Section 3.1.1 is applied for extracting and separating all the possible sections in the report. After this step, the most important keyphrases are extracted using both techniques mentioned in Section 3.1.2: the same TF-IDF model trained for each corresponding section using the labelled reports is applied for detecting general keyphrases, and the IXAMedTagger allows the extraction of additional medical concepts. The lemmatization, stopword removal and bag-of-word transformation techniques aforementioned are applied to each keyphrase in order to ensure that the final format of keyphrases in the test reports is exactly the same as the one that can be found in the knowledge bases.

Once this report processing is complete, keyphrases in the test report are searched within the knowledge bases, and each possible ICD-10 code is weighted by transforming the *p*-value assigned to the pair (ICD-10 code, keyphrase) into a weight directly proportional to the statistical significance indicated by the p-value. This calculation is carried out using the following formula:

$$w = ln\left(\frac{p_0}{p}\right) \tag{5}$$

where *w* is the weight assigned to the ICD-10 code related to a particular keyphrase, p_0 is the threshold considered assuring a 99% statistical significance when building the knowledge bases ($p_0 = 0.01$) and p is the p-value assigned to the pair (ICD-10 code, keyphrase). Hence, the weight of the relation will be proportional to the order-of-magnitude difference between p and p_0.

The final score of an ICD-10 code for a specific test report is the sum of the weights of all the keyphrases appearing in the report that are related to that ICD-10 code in the knowledge base. However, since various sections and their corresponding knowledge bases can be used for calculating the weights of the ICD-10 codes, a method must be defined for determining which p-value is considered when the same keyphrase appears in two or more sections of the report, and is related to the same ICD-10 code in different knowledge bases with different *p*-values. For instance, the keyphrase “*insuficiencia cardiaca*” (“cardiac insufficiency”) could be related to ICD-10 code “I50.9” (see Fig. 3) in the knowledge bases that represent sections “Background and personal family/clinical history” and “Reason for medical consultation”, with p-values p_1 and p_2 and weights w_1 and w_2 respectively. A total score must be eventually assigned to a particular ICD-10 code, based on the keyphrases related to it. When it comes to different keyphrases, it seems intuitive to sum up all their weights to obtain this final score. However, when the same keyphrase presents two or more different possible weights due to having appeared in two or more different sections, it is interesting to explore other possibilities, such as taking the maximum weight (minimum p-value) or the mean of the weights. After some tests, the strategy that sums up all the weights, regardless of the number of sections in which a keyphrase appears, was selected for the final configuration of the system, as it will be explained in Section 4.3.

A final ranking will be hence created for each test report in which all the ICD-10 codes found to be related to the report according to the knowledge bases will be ordered from highest to lowest total weight. Using this ranking, the system will be able to propose as many ICD-10 codes for a particular test report as desired.

4. Experiments and results

In this section the general evaluation framework for the proposed problem is described. The dataset and metrics used for evaluating our system are presented, and the main results achieved by our system are shown. Additionally, we perform a comparison with systems addressing similar tasks, as well as an error analysis of possible gaps in our system. From this analysis, some possible improvements are depicted.

4.1. Dataset

The dataset used for building the knowledge base and performing the different experiments shown in this work consists of 12,966 medical reports from year 2016, containing discharge information from patients leaving the hospital, manually annotated by experts in the field. Reports may contain information about many different hospital services, such as internal medicine, obstetrics and gynecology or cardiology. As a result, the ICD-10 codes related to the reports are very diverse. A subset of 10,118 reports have been used for building the knowledge bases, and the remaining 2,848 reports were reserved for testing purposes. The total number of unique ICD-10 codes in the subset used for building the knowledge bases is 7,592. In particular, an average of 10.37 different ICD-10 codes are assigned to each of the reports used for building the knowledge bases, and 9.46 ICD-10 codes are assigned in average to each report considered for testing purposes. There exist two main categories of ICD-10 codes: diagnosis and procedure codes. However, this differentiation has not been considered for the purposes of this work, and hence all codes are used for classification in a similar way. The average number of characters in the reports used for the experiments described in this work is 7,737.08, while the average number of words per report is 1,471.04.

As mentioned in Section 1, the huge number of possible labels in the classification problem, together with the fact that many different labels can be assigned to a particular report, turns the addressed problem into an extreme multilabel classification task. One of the main characteristics of these tasks, as well as a recurrent problem, is an exceptionally high data imbalance between classes. In general, a small subset of the labels is generally assigned to a large majority of the documents. On the other hand, there usually exists a huge number of labels which appear very rarely as assigned to a particular report. Fig. 4 illustrates this phenomenon.

As it can be observed in the charts, the most frequent labels are assigned to the vast majority of the medical reports, up to the point that the hundred most frequent labels cover around 70% of the considered reports. A closer look at the dataset shows that more than 7,000 labels are assigned to less than 100 medical reports, and around 6,500 labels are assigned to 10 medical reports or less.

In order to better illustrate the results obtained by the proposed system, different testing frameworks have been developed, by considering diverse subsets of labels to be classified. Each framework is built by only considering those labels appearing in more than a specific percentage of the total number of medical reports. Once those labels have been selected, the test set is filtered by only considering those reports presenting at least one of the labels, and hence the total number of possible labels in the task, as well as the average number of labels assigned to each test report in the Gold Standard, is reduced. Table 1 shows the total number of documents in each test set, the total number of classes (labels) in the task and the average number of labels assigned to each test report.

As the minimum percentage of appearance increases, the framework is more restrictive in terms of the number of classes that will be considered, and hence the task difficulty is somehow reduced. In the framework “*Test all*” no restrictions are applied on the minimum appearance of the labels, and hence all the possible documents and classes in the test dataset are considered. In this case, it should be noted that the number of possible labels in the test dataset is smaller than the number of labels used for building the knowledge bases. However, the difference between both sets is not empty, which indicates that there exist classes in the test dataset that have never been seen in the process of building the knowledge base, making it impossible to correctly classify them. In the other frameworks, as we are considering the most frequent labels, all of those labels can be found in the knowledge bases.

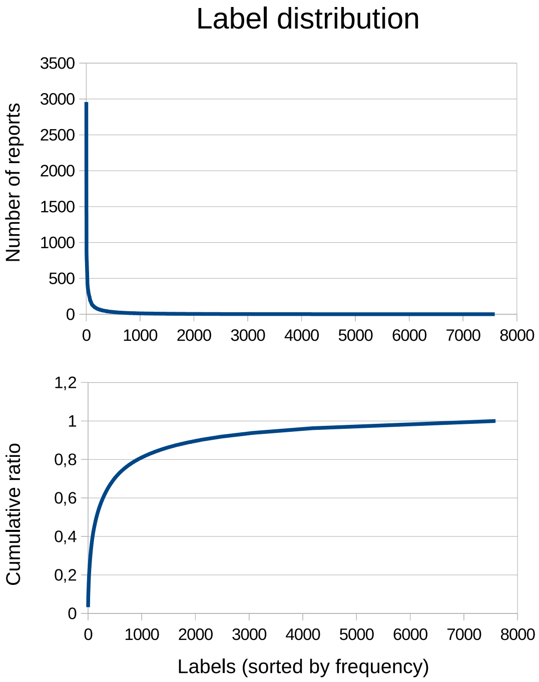


Fig. 4. Distribution of the number of reports. X axis represents labels, sorted by frequency. Above chart shows the number of reports presenting each label. Below chart shows the cumulative distribution function of the total reports covered by the labels.

Table 1
Test frameworks developed from the original dataset. Second column shows the minimum percentage of documents assigned to a label for that label to be included in the test. Third column indicates the total number of test documents and fourth column the total number of labels (classes) considered in the test framework. Finally, the last column shows the average number of labels assigned to each test report in the Gold Standard.

Test framework	Min %	# test docs	# of labels	avg # of labels
Test 5	5%	2,131	21	2.76
Test 4	4%	2,151	27	3.05
Test 3	3%	2,244	45	3.76
Test 2	2%	2,347	76	4.49
Test 1	1%	2,489	175	5.78
Test all	–	2,835	4,089	9.46

4.2. Metrics

Rank-based metrics are commonly used for evaluating extreme multilabel classification tasks [58,59,60], due to the sparsity of positive labels associated to each of the instances to be classified (in this case, medical reports). In this sense, the main metric for assessing the effectiveness of the system is the precision obtained by proposing the K ICD-10 codes with highest weights according to the knowledge bases. This metric, known as *Precision@K* or *P@K*, illustrates the quality of a ranked list of the most important ICD-10 codes emphasising the top portion of the list. However, *F-Measure@K* or *F@K* is also shown in some of the experiments for completeness. The following equations illustrate these metrics:

$$P@K = \frac{1}{K}\sum_{i=1}^{K} r_K(i) \tag{6}$$

$$R@K = \frac{1}{C}\sum_{i=1}^{K} r_K(i) \tag{7}$$

$$F@K = \frac{2P@KR@K}{P@K + R@K} \tag{8}$$

where K is the total number of predicted labels, C is the number of true labels in the Gold Standard, and r_K is a binary vector in which $r_K(i)$ is 1 if the label in position i has been predicted by the system and 0 otherwise. It is important to notice that the maximum value of *Recall@K* (*R@K*) is constrained by K itself, this is, the maximum number of ICD-10 codes proposed by the system. Hence, when the number of possible classes in the Gold Standard, C, is higher than K, *Recall@K* will have a maximum value of $\frac{K}{C}$, which will also affect the final value of *F-Measure@K*, as this metric is the harmonic mean of precision and recall.

A second rank-based metric also widely employed in this kind of tasks has been included in the results: the Normalized Discounted Cumulative Gain (*nDCG@K*). This metric measures the ranking quality by considering the order of the labels in the proposed ranking: correct labels at the top of the ranking list provide higher gain values to the final score than correct labels at the bottom of the ranking list. Normalization of the Discounted Cumulative Gain (DCG) comes from obtaining its ratio in relation to an Ideal DCG (IDCG), through the following formulae:

$$DCG@K = \sum_{i=1}^{K} \frac{rel(i)}{log_2(i+1)} \tag{9}$$

$$IDCG@K = \sum_{i=1}^{|REL_K|} \frac{rel(i)}{log_2(i+1)} \tag{10}$$

$$nDCG@K = \frac{DCG@K}{IDCG@K} \tag{11}$$

where $rel(i)$ is the graded relevance of the result at position i, in this case, represented by the weight assigned to a particular ICD-10 code for the test document, and $|REL_K|$ is the number of true labels in the Gold Standard up to position K.

All evaluation metrics shown in the results have been calculated using micro average values, this is, they have been computed for each test document and then averaged over all the documents in the test dataset.

4.3. Results

General results obtained by the system proposed in this work will be shown in this section in order to illustrate its performance. As it can be deduced by the many considerations taken into account while designing the system (all of them detailed in Section 3), there exist various parameters influencing the final system output. The best configuration found for the system, used for achieving the results shown hereafter, presents the following parameters and heuristics:

- The maximum number of keyphrases considered for each labelled document when building the TF-IDF model is 50, according to the average length of the reports, and the average number of keyphrases that are usually detected.
- The minimum statistical significance threshold is 99%, this is, the maximum *p*-value for which a keyphrase is considered to be linked to an ICD-10 code is 0.01. This statistical significance value has been selected based on the literature, in which a confidence interval of 95–99% is usually recommended. However, since this value is being used for calculating the weight that each keyphrase contributes to the final score, different *p*-values could be used. In that case, keyphrases with high p-values would be transformed into low weights, and hence their contribution to the final score would be negligible.
- After stopword and non-informative keyphrase removal and bag-of-words transformation, there might be pairs (ICD-10 code, keyphrase) presenting the same keyphrase but different p-values. The technique followed in these cases for merging those pairs is to take the minimum p-value of the pair (which will be eventually turned into the maximum weight), as explained in Section 3.3.
- An exhaustive analysis regarding the use of different sections that can be found in the medical reports has been performed for determining those sections that provide the most valuable information for performing the classification. The best results are achieved by combining sections "Background and personal/family clinical history", "Clinical judgment" and "Cardiac risk", although for some of the test frameworks the inclusion of sections "Place of origin header" and "Unidentified section" leads to a slight improvement of the final results. For this reason, we have included the two possible combination of sections in the results shown in the table below.
- As mentioned in Section 3.4, the same keyphrase can be found in more than one section, and hence different weights are to be added to a particular ICD-10 code. Among the different heuristics considered for those cases (maximum weight, sum of weights, average weight), the strategy that sums up all the weights, regardless of whether they come from different keyphrases or from the same keyphrase in different sections, has yielded the best results, although the differences with the other considered heuristics are quite small.

For initially illustrating the performance of the proposed system, Table 2 shows the obtained results when using each of the different sections separately. These results can be seen as a baseline for the system.

As it can be observed, sections "Background and personal/family clinical history" and "Clinical judgment" offer the best results when considered separately, which is consistent with the type of information usually found in those sections. Hence, it is quite likely that any combination of sections that includes them will offer good results.

Table 3 shows the general results obtained by the system proposed in this work, according to the different metrics indicated in Section 4.2, and applied to the test frameworks detailed in Section 4.1. System results are initially compared to a baseline that assigns the K most frequent ICD-10 codes extracted from the labelled reports used for building the

Table 2
Results obtained by the proposed system, when using each section separately for performing the classification. Framework *Test 5* is considered. Last row shows results obtained by the best combination of sections. Scores are shown as a percentage. Bold indicates the best performing sections.

Section	P@1	P@5	P@10	nDCG@5	nDCG@10
Background	**62.98**	**39.78**	**29.94**	**64.30**	**69.97**
Place of origin	38.24	22.05	15.70	38.30	45.01
Application header	0.09	0.12	0.12	0.15	0.18
Patient status	30.50	23.12	21.16	31.50	36.12
Numerical indicators	20.18	12.94	10.77	20.29	25.92
Textual indicators	24.26	15.29	13.52	24.63	32.82
Clinical judgment	**44.91**	**26.69**	**20.09**	**41.98**	**48.34**
Reason for consultation	37.82	24.01	19.34	40.84	50.57
Summary of procedures	35.15	21.66	19.56	36.01	45.31
Cardiac risk	8.78	5.63	4.15	8.28	9.41
Treatment	31.16	21.96	17.73	32.62	40.50
Unidentified	20.51	15.27	13.78	23.80	31.26

Table 3
Results obtained by the proposed system, both when using a combination of 3 sections (“Background and personal/family clinical history”, “Clinical judgment” and “Cardiac risk”) and a combination of 5 sections (the three mentioned sections, together with “Place of origin header” and “Unidentified section”). Results are compared to the “Most Frequent” baseline. Precision is calculated considering 1, 5 or 10 positions of the ranking, and nDCG is shown considering 5 or 10 positions of the ranking. Scores are shown as a percentage. Bold indicates the best result for each metric, in each test framework.

Dataset	System	P@1	P@5	P@10	nDCG@5	nDCG@10
Test 5	Baseline	37.87	22.21	17.1	31.64	37.75
	Proposed (3 s)	74.94	46.43	33.06	75.76	80.93
	Proposed (5 s)	**76.26**	**47.73**	**33.24**	**78.62**	**83.37**
Test 4	Baseline	37.52	22.01	16.94	31.34	37.40
	Proposed (3 s)	74.62	47.36	34.08	73.72	78.67
	Proposed (5 s)	**76.10**	**48.62**	**34.40**	**76.36**	**81.15**
Test 3	Baseline	35.96	21.10	16.24	30.05	35.85
	Proposed (3 s)	72.15	45.68	33.46	68.60	73.07
	Proposed (5 s)	**73.31**	**46.86**	**33.94**	**70.67**	**75.14**
Test 2	Baseline	34.38	20.17	15.53	28.73	34.28
	Proposed (3 s)	**70.09**	46.08	32.69	64.03	68.34
	Proposed (5 s)	67.70	**46.66**	**33.26**	**64.35**	**69.16**
Test 1	Baseline	32.42	19.02	14.64	27.09	32.32
	Proposed (3 s)	**68.58**	**47.05**	33.70	**60.19**	**61.61**
	Proposed (5 s)	62.43	46.28	**34.06**	57.95	60.47
Test all	Baseline	28.47	16.70	12.85	23.78	28.36
	Proposed (3 s)	**63.70**	**44.47**	**32.55**	**50.58**	**46.09**
	Proposed (5 s)	57.43	42.05	31.67	47.26	43.73

knowledge bases.

Our system is able to overcome the Most Frequent baseline in all the cases, achieving higher performance in a consistent way throughout all the test frameworks and metrics. Regarding the proposed test frameworks, the performance of the system is better for those frameworks only considering codes assigned to a high number of medical reports. However, although all the indicators decrease as we introduce less frequent codes, the system is still able to achieve interesting precision scores in all the cases. Finally, the combination of sections “Background and personal/family clinical history”, “Clinical judgment”, “Cardiac risk”, “Place of origin header” and “Unidentified section” offers the best results for those test frameworks that consider fewer possible codes to be assigned, while the inclusion of the last two sections is detrimental to the results as we increase the number of less frequent codes. This performance deterioration of the 5-section configuration reinforces the idea that as the number of ICD-10 codes increases the task becomes more difficult to address, due to the limited occurrence of the new codes in the training dataset. Hence, simpler configurations of the system such as the 3-section combination, are probably less sensitive to the increasing noise introduced by these new codes and therefore more robust for correctly classifying the instances in the test dataset.

For better understanding of the behaviour of the system, Fig. 5 shows the evolution of F-Measure values for each test framework, as the number of labels proposed by the system (K) varies. The 5-section configuration is taken into account for illustrating this behaviour.

The most important aspect that can be extracted from the information in the figure is the relationship between the number of labels for which F-Measure is higher and the average number of labels per document in the test dataset, detailed in the last column of Table 1. As it can be observed, in all the cases these two values are very similar: for frameworks *Test 5* and *Test 4* the highest F-Measure is obtained at $K = 3$, being the average number of labels per document 2.76 and 3.05 respectively. For framework *Test 3*, the average number of labels per document is 3.76 and the highest F-Measure comes at $K = 4$. Frameworks *Test 2* and *Test 1* show the highest F-Measure at $K = 5$ and $K = 6$ respectively, and present an average number of labels per document of 4.49 for *Test 2* and 5.78 for *Test 1*. Finally, the test framework that considers all possible labels, *Test all*, presents an average number of labels per document of 9.46, and the highest F-Measure obtained in this case corresponds to $K = 8$ (although the score is very similar to the one achieved when $K = 9$). This fact indicates that, according to F-Measure, the system is correctly converging to the number of labels that should be proposed for each test document.

One of the most important contributions of the system presented in this work is its high interpretability: for each medical report that needs to be classified, we are able to easily visualise the keyphrases that eventually lead to the selection of each ICD-10 code assigned to the report. Fig. 6 illustrates this behaviour.

From the initial medical report and after applying the complete pipeline described in Section 3, the different proposed ICD-10 codes can be extracted, along with the keyphrases detected in the report and their weights associated to each ICD-10 code. As we can observe in the figure, the same keyphrase named “*Insuficiencia cardíaca*” (“cardiac insufficiency”) can be related to different ICD-10 codes with different weights. In particular, this keyphrase is much more related to the code presenting

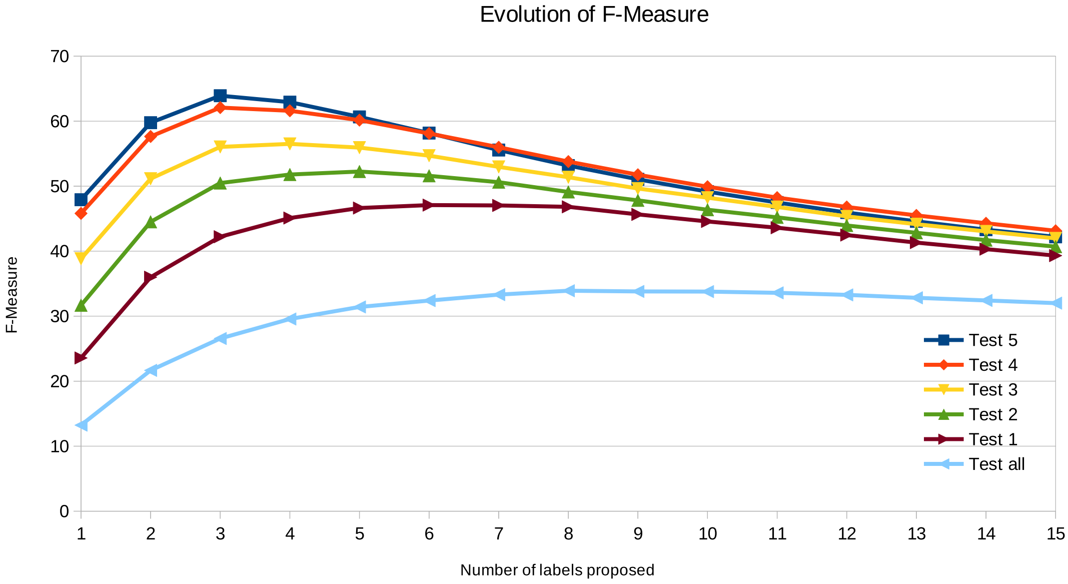


Fig. 5. Evolution of F-Measure@K values obtained by the system, for each of the considered test frameworks. X axis indicates the number of proposed labels (K). Y axis represents F-Measure as a percentage.

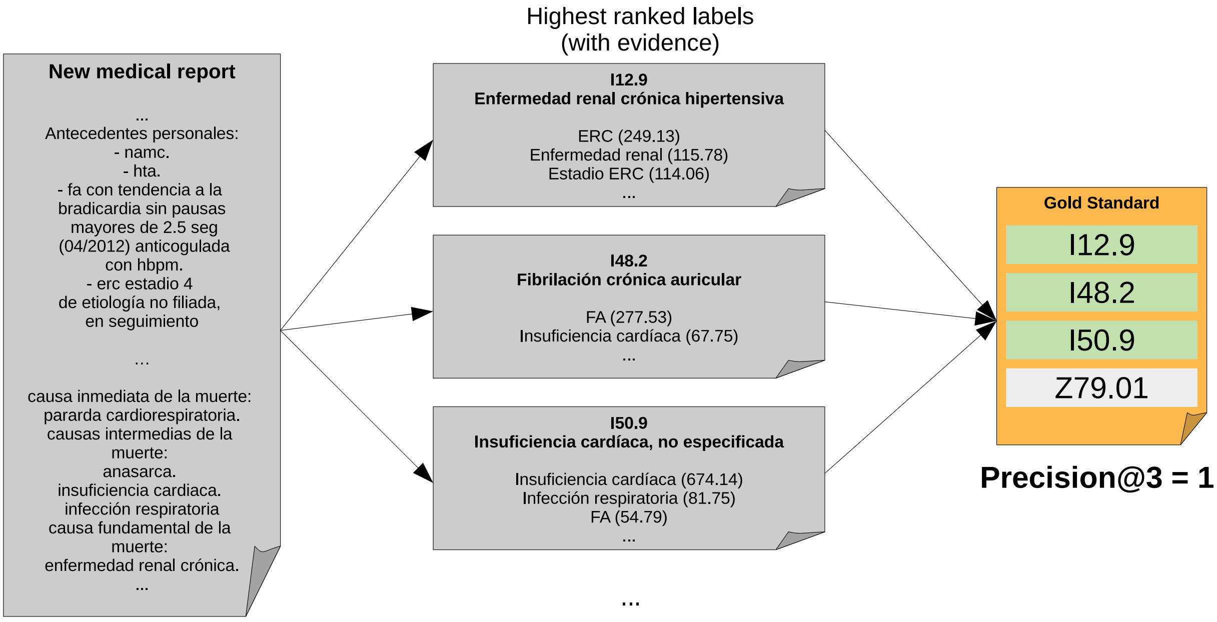


Fig. 6. Example of report classification.

"cardiac insufficiency" in its own description (I50.9) than to I48.2, also associated with cardiac issues but in a different way. The third code (I12.9), used for classifying kidney disorders, obtains its final score from completely different keyphrases also found in the report. Therefore, the three ICD-10 codes found at the top of the ranking are correctly selected for classifying the report, hence returning a Precision@3 value of 1.

Table 4 illustrates the differences between using only the IXA-MedTagger tool for extracting keyphrases from the medical reports, or only the TF-IDF based technique, both explained in Section 3.1.2. The use of both techniques combined is also shown. Results are calculated for the *Test 5* framework and using the combination of 5 sections described before.

Results clearly indicate that combining both IXAMedTagger and TF-IDF methods offers better results than using any of them separately, for all the considered metrics. Regarding F@K, the value of *K* for which the best F-Measure is achieved is consistent with the previous experiment.

Some additional experiments have been performed in order to assess the usefulness of the pre-processing and post-processing techniques employed in the design of the system. In the first experiment, two different methods for extracting the initial candidate keyphrases from the reports are compared: the use of the regular expression detailed in Section 3.1.2 and the use of a constituency parser in the Spanish language for extracting the whole set of noun and prepositional phrases within the text. Table 5 shows the results of both techniques on the *Test 5* framework. The remaining parameters (sections considered, number of keyphrases, statistical significance threshold, etc.) are maintained as before.

The table clearly shows how the use of the proposed regular expression is able to overcome the results obtained when using all noun and prepositional phrases from the reports, for all the considered metrics. This could be due to the higher specificity given by the regular expression, which implicitly filters out some phrases that are not informative enough and hence would introduce noise into the system. This is, the use of the full set of noun and prepositional phrases, although do not necessarily imply introducing completely new expressions, might lead to including redundant keyphrases or less informative keyphrases, hence making it more difficult for the system to determine which ones would be the most important ones for the final classification. In previous works such as [61], the process of automatically extracting keyphrases from text is widely analysed, and it is clearly stated that candidate keyphrases are typically extracted using heuristic rules designed to avoid spurious instances and reduce the number of initial candidates. In the research presented in [62], an initial extraction of candidate keyphrases relies on the use of a regular expression which is actually quite similar to the expression used in this research.

Finally, considering the post-processing step consisting of removing non-informative keyphrases described in Section 3.3, it is important to assess the impact that this post-processing step have on the final results, for illustrating its relevance within the whole system. To this end, the performance of our system when the non-informative keyphrase removal step is excluded from the pipeline is compared to the results when the non-informative keyphrases are indeed removed. The experiment is performed on framework *Test 5*, and the configuration of the parameters is maintained as before, included the combination of 5 sections already explained. Table 6 shows the obtained results.

As indicated by the considered metrics, the impact on performing the mentioned post-processing step is quite significant regarding the overall

Table 4
Results obtained by the proposed system, when using only the IXAMedTagger tool for extracting keyphrases (first row), only the TF-IDF technique (second row) or their combination (third row). Different values of precision and nDCG are considered, as well as the best F-Measure@K (value of K in this case is shown in parentheses). Scores are shown as a percentage. Bold indicates the best result for each metric.

Config.	P@1	P@5	P@10	Best F@K	nDCG@5	nDCG@10
IXA	68.14	42.00	32.58	53.47 (3)	66.44	73.30
TF-IDF	74.33	46.96	33.17	62.70 (3)	76.74	81.81
Both	**76.26**	**47.73**	**33.24**	**63.92 (3)**	**78.62**	**83.37**

Table 5
Results obtained by the proposed system, extracting the initial candidate keyphrases with a regular expression (first row) or using all noun and prepositional phrases in the report (second row). Different values of precision and nDCG are considered. Scores are shown as a percentage. Bold indicates the best result for each metric.

Keyphrase extraction	P@1	P@5	P@10	nDCG@5	nDCG@10
Regular expression	**76.26**	**47.73**	**33.24**	**78.62**	**83.37**
Noun and prep. phrases	68.32	41.53	27.03	73.30	79.28

Table 6
Results obtained by the proposed system, without performing the non-informative keyphrase removal as a post-processing step (first row), or performing the removal (second row). Different values of precision and nDCG are considered. Scores are shown as a percentage. Bold indicates the best result for each metric.

Config.	P@1	P@5	P@10	nDCG@5	nDCG@10
No removal	68.28	38.78	28.14	69.72	77.13
Key removal	**76.26**	**47.73**	**33.24**	**78.62**	**83.37**

performance of the system. Scores for all the metrics are higher when the non-informative keyphrases are removed, which confirms that the process is an important step to include in the general pipeline. Hence, the systematisation of this step will be of crucial importance in future versions of the proposed system.

4.4. System comparison

Given the novelty of the research on automatic ICD-10 code classification in Spanish medical reports, and the general lack of resources for this purpose in the Spanish language, few works can be found in the literature with which the results of our system could be compared in a fair and reliable manner. The main difficulty is the use of different datasets, expensive to generate due to the privacy restrictions they must comply with, which makes most of the datasets very difficult to be shared. The most similar works to which our results could be somehow compared, are those presented in [63] and [60]. The first work, developed by the IXA group, presents a Deep Learning approach consisting of a Bidirectional Recurrent Neural Network for tackling the problem of multilabel classification of ICD-10 codes in Spanish EHRs, exploring different embedding techniques for representing text within the reports. The dataset used in this case contains similar information to the one explored in this research, although it comes from a different source. It follows a comparable distribution in terms of total number of documents, and average number of labels per document, although the average length of the considered documents is significatively smaller (~770 words against ~1470 words). Only diagnoses are taken into account, leaving the procedure codes unused. Some similarities with the present research can be found in that work: different results are provided regarding either the use of the whole document or just the “diagnostic” part. Different granularities are also explored in terms of only considering the first letter of the ICD-10 codes (most general family of diagnoses), just the three first characters of the code representing a so called “block”, or the full code. In all the presented experiments, only those codes appearing in more than 5% of the training dataset are considered. Only general precision, recall and F-score metrics are presented, without considering ranking metrics. Therefore the fairest comparison could be done by considering just our *Test 5* framework (21 labels) against their “full code” setting (16 labels), and considering our highest F-Measure score, achieved for $K = 3$ (see Fig. 5). With those considerations in mind, their best F-Measure score is F = 54.30% when considering the diagnostic section and F = 63.16% when considering the whole document. Our results for the selected configuration achieve F = 63.92%, outperforming their score. Moreover, one of the most important contributions of our system is the interpretability of the obtained results, through the analysis of keyphrases as an evidence of the weight assigned to a particular ICD-10 code for a test document. On the other side, the work by IXA is based on Deep Learning techniques, for which interpretability is much more compromised.

The second comparable research presents a comparison of many different methods for addressing the same extreme multilabel classification problem [20]. Techniques specifically developed for tackling XMLC tasks are explored, as well as Support Vector Machines (SVMs), Multilayer Perceptrons (MLPs) or boosting methods (AdaBoost and GBoost), and also Deep Learning techniques such as Long Short-Term Memory networks (LSTMs) and Convolutional Neural Networks (CNNs). Different features including bag of words, TF-IDF statistics and embeddings are considered for representing the documents. The dataset used for evaluation is similar to the one used in this research, although the total number of documents used for training and test is smaller (around 5,800 documents for training and around 1,450 for test). Metrics including Precision@K and nDCG@K are presented, which makes the comparison with our system easier. The technique offering best results is gradient boosting, obtaining the following scores when considering all the ICD-10 codes: P@1 = 69.47%, P@10 = 40.88% and nDCG@10 = 78.44. These scores outperform the system presented in this work for the whole test dataset (see row *Test all* in Table 3), although our results are comparatively competitive for most of the explored techniques when it comes to considering the most frequent ICD-10 codes (our *Test 5* framework). In that work, a final ensemble technique combining different methods is able to improve these results up to P@1 = 73.53%, P@10 = 41.73% and nDCG@10 = 80.80%, which gives some clues about the need of combining different approaches when considering codes with different frequencies of occurrence. In a similar way with the aforementioned work by the IXA group, an important added value within our research is the direct interpretability of the results offered by the system.

4.5. Error analysis

Some of the medical reports in the test dataset that were not correctly classified in the proposed experiments have been analysed in depth in order to determine the kind of errors the system is producing. This way, some corrective actions can be established and eventually implemented to address these errors. The different detected errors and their corresponding possible solutions are described below:

- **Comorbidities and codes with similar keyphrases**: Comorbidity are defined as the simultaneous presence of two or more diseases or medical conditions in a patient. In our corpus, comorbidity becomes an issue when each disease is associated with a different ICD-10 code. In many of those cases, both diseases and their associated keyphrases will appear in the same report, which will be related to two different ICD-10 codes. In that scenario, the system will probably learn to assign both ICD-10 codes whenever any of those keyphrases are found in a new report. However, if only one of the diseases appears separately within a report, it will be difficult for the system to determine which of the two possible ICD-10 codes is associated with it. This is, reports with only one of the ICD-10 codes in the Gold Standard will be much more difficult to classify in which could be considered an ambiguity problem. In order to further explore this phenomenon, we have performed a deeper analysis of pairs of ICD-10 codes being frequently assigned together to the same report, since in those cases all the keyphrases of the report will be equally related to both ICD-10 codes.

  Different casuistries can be found in the considered corpus: for instance, there is no ambiguity problems when considering comorbidities between “Cardiac insufficiency‘(“*Insuficiencia cardiaca*”), with ICD-10 code I50.9 and “Atrial fibrillation” (“*Fibrilación auricular*”), with code I48.91. When considering the dataset used for building the knowledge base, code I48.91 occurs 336 times, while code I50.9 appears in 730 documents. However, the number of co-occurrences is 107, which is smaller than the number of times each code occurs separately. Therefore the system should be able to correctly discriminate between those two codes. Indeed, when considering the *Test 1* framework, 150 documents are found in the Gold Standard with only code I50.9 assigned (not code I48.91). The system is able to correctly classify 132 out of those 150 documents (88%), this is, by assigning code I50.9 and not assigning code I48.91, when proposing 6 codes per document, which is the average number of codes in the *Test 1* framework, as explained in Section 4.1.

However, cases in which somehow unrelated ICD-10 codes usually co-occur in the same reports can be found within the dataset, although they do not necessarily represent comorbidity situations. A good example is the pair of frequently co-occurring codes N39.0 ("urinary tract infection") and B96.1 ("*Klebsiella pneumoniae* as cause of diseases incorrectly classified"). These codes are actually related since "*K. pneumoniae*" is a bacterium that may cause urinary tract infections. Both codes appear together in 106 documents of the dataset used for building the knowledge base, from a total of 137 occurrences of code B96.1 and 738 occurrences of code N39.0. This implies that, especially for code B96.1, the number of co-occurrences with code N39.0 is much higher than the number of isolated occurrences. In this case, for the *Test 1* framework, the system is able to identify 20 out of 40 test cases with both codes in the gold standard (50%), due to the high co-occurrence in the training dataset. On the other hand, out of 19 test documents with code B96.1 assigned but missing code N39.0, only 1 of them is correctly detected by the system (5.26%), this is, classified with code B96.1 and not with code N39.0. The system incorrectly assigns both codes in 7 out of those 19 cases (36.84%). This shows how, when codes (and therefore their associated keyphrases) frequently appear together within documents in the training dataset, the system usually learns this relation and hence finds it quite harder to correctly differentiate them when it is needed.

One of the possible strategies that could be adopted for reducing the impact of this issue is related to the same statistical process that is conducted for associated each keyphrase to a particular ICD-10 code, described in Section 3.2. This process tends to assign smaller weights to pairs of elements frequently co-occurring, especially if the number of co-occurrences is much higher than the number of isolated occurrences of each element separately. As an example, if two keyphrases co-occurred in 18 documents out of a dataset of 20 documents, and both of them appeared in 19 documents each (this is, both keyphrases would appear in just one document in a separate way), their associated *p*-value would be around 0.05, which would be even higher than the confidence threshold of 99% considered in this work for creating a link between the two elements. The proposed strategy, hence, could be to perform an additional statistical analysis considering pairs of keyphrases in addition to pairs (keyphrase, ICD-10 code), which would ideally allow us to filter out those pairs of keyphrases that, being associated to two different ICD-10 codes, also present a weak co-occurrence relation, represented by a small weight or a high p-value. This way we could remove ambiguous pairs of keyphrases from those ICD-10 codes sharing them, and focus on keyphrases that are able to represent ICD-10 codes in a unique way.

- **Labels not appearing in the Gold Standard**: Some of the detected errors correspond to test reports in which enough information can be found which would lead to label the report with a particular ICD-10 code, however, this label is not found in the Gold Standard. For instance, a specific case can be found within the test dataset in which a history of cardiac insufficiency ("*insuficiencia cardíaca*" and its acronyms, "*IC*" and "*ICC*") is clearly reported. Due to this evidence, the ICD-10 code with higher weight among those proposed by the system is I50.9 ("unspecified cardiac insufficiency"). Nevertheless, this particular label cannot be found in the Gold Standard, and hence $P@1 = 0$ for this report. These types of errors usually affect more to values of $P@K$ for which K is small, since normally the system is able to find other codes actually present in the Gold Standard, although they are ranked lower. No straightforward solution could be proposed for solving these errors, apart from reviewing the Gold Standard for avoiding them.
- **Negation-related errors**: The effect of negation plays an important role in the whole process. Although keyphrases containing negation triggers in Spanish such as "*no*", "*ni*" or "*sin*" are usually properly located thanks to the regular expressions described in Section 3.1.2, the scope of those triggers is not always correctly detected, which affects to the extraction of correct keyphrases. For instance, a test report contains the sentence "*No HTA*, *DM ni DL*". In this case, the acronyms correspond to "high blood pressure", "diabetes mellitus" and "hyperlipidemia", respectively. However, since the scope of the negation is not correctly detected, keyphrases such as "*HTA*", "*DM*" or "*DL*" are found to be related to the report. This causes the system to assign ICD-10 codes related to those three conditions in a case in which none of those codes should be assigned. Indeed, when removing these three ICD-10 codes from the ranking proposed by the system, the following code with highest weight is effectively found in the Gold Standard, which indicates that a better management of negation triggers and scopes would lead to a correct classification in these cases.
- **Differences between diagnoses and procedure codes**: Some of the errors detected in the proposed system could be avoided by separately considering diagnoses and procedure ICD-10 codes. More specifically, an analysis of the sections within the medical reports to be considered for assigning the two different types of codes should be conducted. A separate evaluation of the classification of the two types of codes should also be proposed in order to properly analyse the accuracy of the system in relation to this issue.
- **Low frequency codes**: The sparse distribution of labels, which is characteristic of the kind of problem being addressed in this work, also generates a particular type of errors. There will be many cases in which it will be impossible to gather the amount of information that is needed for assigning keyphrases to particular ICD-10 codes in a robust and accurate way. This fact occurs when the ICD-10 code appears very rarely, or does not appear at all, in the training dataset. This fact can be clearly seen when observing the most significant keyphrases associated to frequently appearing codes, and comparing them to those main keyphrases associated to rare codes. For instance, code *E11.9*, with description "*Diabetes mellitus tipo 2*, *sin complicaciones*" ("Type 2 diabetes mellitus, without complications") is one of the most frequently appearing codes in the training dataset. The keyphrases with lowest *p*-values (and hence with highest weights) associated to that code are "*diabetes*", "*diabetes mellitus*", "*dm 2*" or "*dm II*". Moreover, the associated p-values are really small, which will lead to very high weights and to almost certainly assign that code in case any of those keyphrases is found in a test report. On the other side, if we take code *D13.2*, with description "*Neoplasia benigna de duodeno* ("Benign duodenal neoplasia"), which appear in less than 1% of the reports, we will find unrelated keyphrases such as "*prótesis aórtica biológica*" ("biological aortic prosthesis") or "*portador de marcapasos*" ("cardiac pacemaker wearer"). In addition, the weights associated to those keyphrases will be similar or even higher than those associated to other keyphrases somehow more related to the code, such as "*hemorragia digestiva*" ("digestive bleeding") or "*biopsia de pólipo duodenal*" ("duodenal polyp biopsy"). Again, the solution for these kind of errors begins by collecting a greater amount of information for enriching the knowledge bases and better discriminating between ICD-10 codes. Additional annotation is being currently carried out on new medical reports with similar characteristics, hence we hope to be able to work with a wider dataset in future experiments.
- **Unremoved non-informative keyphrases**: Given that the post-processing step consisting of removing non-informative keyphrases is performed manually, it is clear that there will exist keyphrases that are not introducing any useful information into the system, but are not being removed because they intuitively seem to be important. For instance, we can find keyphrases that, even providing information, might appear in many different contexts and hence be related to many different ICD-10 codes. In those cases, the informativeness of the keyphrase would be drastically reduced given its low capacity to have any influence when it comes to discriminate between ICD-10 codes to assign to a report. This could be seen as a domain specific stopword detection and removal, that is, locating particular words

and phrases that, although not considered to be general stopwords in the considered language, behave as such in specific domains and contexts. Therefore, some modifications should be added in the statistical analysis of the reports in order to detect those stopwords.

- **Undetected informative keyphrases**: Finally, there exist errors in the classification that might be due to the incorrect removal of keyphrases that could be useful for detecting particular ICD-10 codes. This removal could be performed both in the TF-IDF model calculation described in Section 3.1.2 and in the statistical process explained in Section 3.2. Since both processes rely on the frequency of occurrence and co-occurrence of keyphrases, gathering more data will be again an important issue to be taken into account in order to avoid this type of errors, which affect to the global recall of the system. As mentioned before, these errors can be usually detected related to those codes appearing rarely in the training dataset.

5. Conclusions and future work

This paper presents a novel and highly interpretable technique for addressing the task of ICD-10 code classification of Spanish medical reports. The task can be addressed as an extreme multilabel classification problem, given the high number of classes, represented by all the possible ICD-10 codes that can be assigned to a particular medical report. A complete pipeline is presented in this work for processing the available reports, extracting knowledge from them and applying this knowledge for classifying new unlabelled reports. The knowledge base creation is focused on extracting statistically significant relations between keyphrases found in the reports and the ICD-10 codes assigned to those reports. This information is subsequentially used for weighting the candidate ICD-10 codes to be assigned to a new unlabelled report. The obtained results particularly indicates the usefulness of the generated knowledge bases for interpreting medical reports in terms of important keywords and keyphrases that eventually lead to a correct classification. As shown in Fig. 3, the most important keyphrases associated to the ICD-10 codes are semantically very close to their descriptions, which shows that this semantic knowledge can be effectively extracted from medical reports written using natural language, which in some cases might be even informal and inaccurate.

Some different and very interesting future lines of work have been depicted along this paper, and should be addressed for improving the quality of the proposed system and achieving better performance in the classification task: various techniques for enhancing the extraction of keyphrases in the medical reports could be explored. Also, techniques for detecting keyphrase variants could be analysed in order to increase the coverage of the system. These variants may be applied at the lexical level for solving writing errors which are common in non-standard and informal reports, but also at syntactic and semantic levels for enlarging the amount of different expressions related to a same concept that can be detected by the system. The segmentation utility used for detecting different sections in the reports should also be subject to further analysis and improvement. Moreover, detection and removal of non-informative keyphrases in the post-processing steps has shown to be a key step for boosting the global results, and hence the automatisation of this process might offer interesting improvements on the system.

The interpretability of the results offered by the system allows us to explore those cases in which the proposed technique is not working accurately, as shown in Section 4.5. From those analysis we can foresee additional future lines of work such as the effect of negation triggers and scopes, improvements on the Gold Standard, or the separate analysis of diagnoses and procedure codes.

The comparison with other systems presented in Section 4.4 also gives us clues about possible research directions to be explored. For instance, the combination of the method proposed in this work with other machine learning techniques could be a promising source of improvement.

Data availability

The dataset used in this research has restrictions of use due to the European General Data Protection Regulation (EU GDPR), so it cannot be made public for the research community, even if anonymised.

Declaration of competing interest

The authors declare no conflict of interest regarding this manuscript submitted to Artificial Intelligence in Medicine.

Acknowledgments

This work has been partially supported by the Spanish Ministry of Science and Innovation within the DOTT-HEALTH Project (MCI/AEI/FEDER, UE) under Grant PID2019-106942RB-C32, as well as project EXTRAE II (IMIENS 2019) and the research network AEI RED2018-102312-T (IA-Biomed).

Appendix A Sections extracted with the segmentation utility

- ***Antecedentes e historia clínica personal/familiar***: Background and personal/family clinical history.
- ***Cabecera de procedencia***: Place of origin header.
- ***Cabecera de solicitud***: Application header.
- ***Datos de situación del paciente***: Patient status data.
- ***Estudios de indicación exclusivamente numérica***: Studies showing purely numerical indicators
- ***Estudios de indicación textual***: Studies showing textual indicators
- ***Juicio clínico***: Clinical judgment.
- ***Motivo de consulta***: Reason for medical consultation.
- ***Resumen de actuaciones***: Summary of medical procedures.
- ***Riesgo cardíaco***: Cardiac risk.
- ***Tratamiento***: Treatment.
- ***Sección sin identificar***: Unidentified section.

Appendix B Non-informative keyphrases

“Abandono”, “acorde”, “actividad”, “actual seguimiento”, “actualidad”, “ajuste nuevo”, “alerta”, “ambulatorio seguimiento”, “animal”, “antecedente”, “antecedente descrito”, “antecedente edad”, “antecedente familiar”, “artefacto”, “cambio necesidad”, “campo”, “causa posible”, “cena”, “cita”, “cita posterior”, “colección”, “comentario”, “comentario mujer”, “comida”, “conclusión”, “consulta”, “consulta ambulatorio seguimiento”, “consulta cura”, “consulta externo”, “consulta medicina interno”, “consulta seguimiento”, “contacto”, “contacto primer”, “contexto”, “cuidado”, “cuidadora principal”, “desayuno”, “diagnostico”, “domiciliario seguimiento”, “domicilio”, “factor”, “familia”, “forma”, “general estado”, “habitual especialista”, “habitual medico”, “historia”, “hospital”, “hospital ingreso”, “hospital referencia”, “hospitalario ingreso”, “incidencia”, “informe”, “informe alto medicina interno”, “ingreso”, “ingreso actual”, “ingreso anterior”, “ingreso reciente”, “ingreso urgencia”, “interconsulta”, “interno medicina ingreso”, “interno medicina planta”, “juicio”, “lenguaje”, “manera indefinido”, “medicina”, “medicina interno”, “mixto”, “modificación presente”, “momento”, “motivo alto fin”, “nota”, “nota interconsulta”, “origen”, “parte”, “pauta”, “periodo”, “persona”, “plan alto seguimiento”, “planta”, “posterior revisión”, “presente”, “procedimiento”, “propio”, “prueba”, “punto vista”, “recomendación”, “respuesta”, “resultado”, “seguimiento”, “semana”, “semana fin”, “servicio”, “soporte habitual”, “territorio”, “traslado”, “unidad”, “valoracion”.

References

[1] M. CodeBooks. ICD-10-CM Complete Code Setvol. 1. Medical Code Books; 2016. p. 2016. URL: https://www.medicalcodebooks.com/coding-books/icd-10-cm-complete-code-set.

[2] Liu J, Chang W-C, Wu Y, Yang Y. Deep learning for extreme multi-label text classification. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval; 2017. p. 115–24.

[3] Mujtaba G, Shuib L, Raj RG, Rajandram R, Shaikh K, Al-Garadi MA. Automatic icd-10 multi-class classification of cause of death from plaintext autopsy reports through expert-driven feature selection. PLoS One 2017;12:1–27. URL: https://doi.org/10.1371/journal.pone.0170242.

[4] Atutxa A, Casillas A, Ezeiza N, Fresno V, Goenaga I, Gojenola K, et al. Ixamed at CLEF ehealth 2018 task 1: ICD10 coding with a sequence-to-sequence approach. In: Cappellato L, Ferro N, Nie J, Soulier L, editors. Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, Avignon, France, September 10–14, 2018. volume 2125. CEUR Workshop Proceedings, CEUR-WS.org; 2018. of.

[5] Holzinger A, Langs G, Denk H, Zatloukal K, Müller H. Causability and explainabilty of artificial intelligence in medicine. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 2019;9:e1312. https://doi.org/10.1002/widm.1312.

[6] Xu K, Lam M, Pang J, Gao X, Band C, Mathur P, et al. Multimodal machine learning for automated icd coding. In: Doshi-Velez F, Fackler J, Jung K, Kale D, Ranganath R, Wallace B, et al., editors. Proceedings of the 4th Machine Learning for Healthcare Conference. Vol. 106. Ann Arbor, Michigan: Proceedings of Machine Learning Research, PMLR; 2019. p. 197–215. of.

[7] Johnson A, Pollard T, Shen L, Lehman L-w, Feng M, Ghassemi M, et al. Mimic-iii, a freely accessible critical care database. Scientific Data 2016;3:160035. https://doi.org/10.1038/sdata.2016.35.

[8] Nguyen AN, Truran DL, Kemp M, Koopman B, Conlan D, O'Dwyer J, et al. Computer-assisted diagnostic coding: Effectiveness of an nlp-based approach using SNOMED CT to ICD-10 mappings. In: AMIA 2018, American Medical Informatics Association Annual Symposium, San Francisco, CA, November 3–7, 2018; 2018. p. 807–16.

[9] Donnelly K. Snomed-ct: the advanced terminology and coding system for ehealth. Stud Health Technol Inform 2006;121:279.

[10] Aronson AR. Effective mapping of biomedical text to the UMLS metathesaurus: the MetaMap program. In: Proceedings of AMIA, Annual Symposium; 2001. p. 17–21.

[11] Chapman WW, Bridewell W, Hanbury P, Cooper GF, Buchanan BG. A simple algorithm for identifying negated findings and diseases in discharge summaries. J Biomed Inform 2001;34:301–10.

[12] Humphreys BL, Lindberg DA, Schoolman HM, Barnett GO. The unified medical language system: an informatics research collaboration. J Am Med Inform Assoc 1998;5:1–11.

[13] Subotin M, Davis A. A system for predicting ICD-10-PCS codes from electronic health records. In: Proceedings of BioNLP 2014. Baltimore, Maryland: Association for Computational Linguistics; 2014. p. 59–67. URL: https://www.aclweb.org/anthology/W14-3409. https://doi.org/10.3115/v1/W14-3409.

[14] Névéol A, Grouin C, Cohen KB, Hamon T, Lavergne T, Kelly L, et al. Clinical information extraction at the CLEF eHealth Evaluation Lab 2016. In: Proc of CLEF eHealth Evaluation Lab, Evora, Portugal; 2016. p. 28–42.

[15] Névéol A, Robert A, Anderson R, Cohen KB, Grouin C, Lavergne T, et al. CLEF ehealth 2017 multilingual information extraction task overview: ICD10 coding of death certificates in English and French. In: Cappellato L, Ferro N, Goeuriot L, Mandl T, editors. Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum, Dublin, Ireland, September 11–14, 2017. volume 1866. CEUR Workshop Proceedings, CEUR-WS.org; 2017. of.

[16] Névéol A, Robert A, Grippo F, Morgand C, Orsi C, Pelikan L, et al. CLEF ehealth 2018 multilingual information extraction task overview: ICD10 coding of death certificates in french, hungarian and italian. In: Cappellato L, Ferro N, Nie J, Soulier L, editors. Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, Avignon, France, September 10–14, 2018. volume 2125. CEUR Workshop Proceedings, CEUR-WS.org; 2018. of.

[17] Seva J, Sänger M, Leser U. WBI at CLEF ehealth 2018 task 1: language-independent ICD-10 coding using multi-lingual embeddings and recurrent neural networks. In: Cappellato L, Ferro N, Nie J, Soulier L, editors. Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, Avignon, France, September 10–14, 2018. volume 2125. CEUR Workshop Proceedings, CEUR-WS.org; 2018. of.

[18] Jeblee S, Budhkar A, Milic S, Pinto J, Pou-Prom C, Vishnubhotla K, et al. Toronto CL CLEF 2018 ehealth task 1: Multi-lingual ICD-10 coding using an ensemble of recurrent and convolutional neural networks. In: Cappellato L, Ferro N, Nie J, Soulier L, editors. Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, Avignon, France, September 10–14, 2018. volume 2125. CEUR Workshop Proceedings, CEUR-WS.org; 2018. of.

[19] Ive J, Viani N, Chandran D, Bittar A, Velupillai S. Kcl-health-nlp@clef ehealth 2018 task 1: ICD-10 coding of french and italian death certificates with character-level convolutional neural networks. In: Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, Avignon, France, September 10–14, 2018; 2018.

[20] Almagro M, Martínez R, Montalvo S, Fresno V. A cross-lingual approach to automatic icd-10 coding of death certificates by exploring machine translation. J Biomed Inform 2019;94:103207. https://doi.org/10.1016/j.jbi.2019.103207.

[21] Atutxa A, de Ilarraza AD, Gojenola MO koldo, de Viñaspre OP. Interpretable deep learning to map diagnostic texts to icd10 codes. Int J Med Inform 2019. https://doi.org/10.1016/j.ijmedinf.2019.05.015. Link to publication: https://authors.elsevier.com/c/1ZANI4xGJ~syOE.

[22] Miranda-Escalada A, Gonzalez-Agirre A, Armengol-Estapé J, Krallinger M. Overview of automatic clinical coding: annotations, guidelines, and solutions for non-english clinical cases at codiesp track of clef ehealth 2020. In: Working Notes of Conference and Labs of the Evaluation (CLEF) Forum CEUR Workshop Proceedings; 2020.

[23] Blanco A, Pérez A, Casillas A. Ixa-aaa at clef ehealth 2020 codiesp. In: Automatic Classification of Medical Records With Multi-label Classifiers and Similarity Match Coders. in: CLEF; 2020.

[24] Cossin S, Jouhet V. IAM at CLEF eHealth 2020: concept annotation in Spanish electronic health records. In: Working Notes of Conference and Labs of the Evaluation (CLEF) Forum. CEUR Workshop Proceedings; 2020.

[25] García-Santa N, Cetina K. FLE at CLEF ehealth 2020: text mining and semantic knowledge for automated clinical encoding. In: Cappellato L, Eickhoff C, Ferro N, Névéol A, editors. Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, Thessaloniki, Greece, September 22–25, 2020. volume 2696. CEUR Workshop Proceedings, CEUR-WS.org; 2020. of. URL: http://ceur-ws.org/Vol-2696/paper_111.pdf.

[26] Ning W, Yu M, Zhang R. A hierarchical method to automatically encode Chinese diagnoses through semantic similarity estimation. BMC Med Inform Decis Mak 2016;16:30. URL: https://doi.org/10.1186/s12911-016-0269-4.

[27] Dong Z, Dong Q. Hownet - a hybrid language and knowledge resource. In: International Conference on Natural Language Processing and Knowledge Engineering, 2003. Proceedings 2003. IEEE; 2003. p. 820–4.

[28] Chen Y, Lu H, Li L. Automatic ICD-10 coding algorithm using an improved longest common subsequence based on semantic similarity. PLoS One 2017;12:1–17. URL: https://ideas.repec.org/a/plo/pone00/0173410.html. https://doi.org/10.1371/journal.pone.0173.

[29] Almagro-Cádiz M, Martínez R, Fresno V, Montalvo S. Estudio preliminar de la anotación automática de códigos CIE-10 en informes de alta hospitalarios. Procesamiento del Lenguaje Natural 2018;60:45–52.

[30] Merrouni ZA, Frikh B, Ouhbi B. Automatic keyphrase extraction: a survey and trends. J Intell Inf Syst 2020;54:391–424.

[31] Papagiannopoulou E, Tsoumakas G. A review of keyphrase extraction. Wiley Interdiscip Rev Data Min Knowl Discov 2020;10.

[32] Frank E, Paynter GW, Witten IH, Gutwin C, Nevill-Manning CG. Domain-specific keyphrase extraction. In: Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, IJCAI '99. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 1999. p. 668–73.

[33] Witten IH, Paynter GW, Frank E, Gutwin C, Nevill-Manning CG. Kea: Practical automatic keyphrase extraction. In: Proceedings of the Fourth ACM Conference on Digital Libraries DL '99. New York, NY, USA: Association for Computing Machinery; 1999. p. 254–5.

[34] Mihalcea R, Tarau P. Textrank: Bringing order into text. In: Proceedings of the 2004 conference on empirical Methods in Natural Language Processing, EMNLP 2004, a meeting of SIGDAT, a Special Interest Group of the ACL, Held in Conjunction with ACL 2004, 25–26 July 2004. Barcelona, Spain: ACL; 2004. p. 404–11.

[35] Martínez-Romo J, Araujo L, Duque A. Semgraph: extracting keyphrases following a novel semantic graph-based approach. JASIST 2016;67:71–82. URL: https://doi.org/10.1002/asi.23365.

[36] Yu Y, Ng V. Wikirank: Improving unsupervised keyphrase extraction using background knowledge. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7–12, 2018; 2018. p. 3723–7.

[37] Patel K, Caragea C. Exploring word embeddings in crf-based keyphrase extraction from research papers. In: Proceedings of the 10th International Conference on Knowledge Capture. New York, NY, USA: K-CAP '19, Association for Computing Machinery; 2019. p. 37–44.

[38] Zhang Y, Liu H, Wang S, Ip W, Wei F, Xiao C. Automatic keyphrase extraction using word embeddings. Soft Comput 2019:1–16.

[39] Sarkar K. Automatic keyphrase extraction from medical documents. In: Chaudhury S, Mitra S, Murthy CA, Sastry PS, Pal SK, editors. Pattern Recognition and Machine Intelligence. Berlin, Heidelberg: Springer Berlin Heidelberg; 2009. p. 273–8.

[40] Pomares-Quimbaya A, Kreuzthaler M, Schulz S. Current approaches to identify sections within clinical narratives from electronic health records: a systematic review. BMC Med Res Methodol 2019;19:1–20.

[41] Schuemie MJ, Trieschnigg D, Meij E. Dutchhattrick: Semantic query modeling, context, section detection, and match score maximization. In: Voorhees EM, Buckland LP, editors. Proceedings of The Twentieth Text REtrieval Conference, TREC 2011, Gaithersburg, Maryland, USA, November 15–18, 2011. volume 500–296. NIST Special Publication, National Institute of Standards and Technology (NIST); 2011. of. URL: http://trec.nist.gov/pubs/trec20/papers/DutchHatTrick.med.update.pdf.

[42] Singh M, Murthy A, Singh S. Prioritization of free-text clinical documents: a novel use of a Bayesian classifier. JMIR Med Inform 2015;3:e17.

[43] Meystre S, Haug PJ. Automation of a problem list using natural language processing. BMC Med Inform Decis Mak 2005;5:1–14.

[44] Ramos J, et al. Using tf-idf to determine word relevance in document queries. In: Proceedings of the First Instructional Conference on Machine Learning. 242; 2003. p. 133–42. New Jersey, USA.

[45] Schmid H. Probabilistic part-ofispeech tagging using decision trees. In: New Methods in Language Processing; 2013. p. 154.

[46] Loper E, Bird S. Nltk: the natural language toolkit. In: arXiv Preprint cs/0205028; 2002.

[47] Gojenola K, Oronoz M, Pérez A, Casillas A, Taldea I. Ixamed: Applying freeling and a perceptron sequential tagger at the shared task on analyzing clinical texts. SemEval@ COLING; 2014. p. 361–5.

[48] Casillas A, de Ilarraza AD, Fernandez K, Gojenola K, Oronoz M, Pérez A, et al. Ixamed-ie: on-line medical entity identification and adr event extraction in Spanish. In: 2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). IEEE; 2016. p. 846–9.

[49] Casillas A, Gojenola K, Pérez A, Oronoz M. Clinical text mining for efficient extraction of drug-allergy reactions. In: 2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). IEEE; 2016. p. 946–52.

[50] Perez A, Weegar R, Casillas A, Gojenola K, Oronoz M, Dalianis H. Semi-supervised medical entity recognition: a study on Spanish and Swedish clinical corpora. J Biomed Inform 2017;71:16–30.

[51] L. Padró, Reese S, Agirre E, Soroa A. Semantic services in freeling 2.1: Wordnet and ukb. In: 5th Global WordNet Conference; 2010. p. 99–105.

[52] Oronoz M, Casillas A, Gojenola K, Perez A. Automatic annotation of medical records in spanish with disease, drug and substance names. In: Iberoamerican Congress on Pattern Recognition. Springer; 2013. p. 536–43.

[53] Freund Y, Schapire RE. Large margin classification using the perceptron algorithm. Mach Learn 1999;37:277–96.

[54] Collins M. Discriminative training methods for hidden markov models: theory and experiments with perceptron algorithms. In: Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002); 2002. p. 1–8.

[55] Martinez-Romo J, Araujo L, Borge-Holthoefer J, Arenas A, Capitán JA, Cuesta JA. Disentangling categorical relationships through a graph of co-occurrences. Phys Rev E 2011;84:046108.

[56] Duque A, Araujo L, Martinez-Romo J. Co-graph: a new graph-based technique for cross-lingual word sense disambiguation. Nat Lang Eng 2015;21:743.

[57] Duque A, Stevenson M, Martinez-Romo J, Araujo L. Co-occurrence graphs for word sense disambiguation in the biomedical domain. Artif Intell Med 2018;87:9–19.

[58] Bhatia K, Jain H, Kar P, Varma M, Jain P. Sparse local embeddings for extreme multi-label classification. In: Advances in Neural Information Processing Systems; 2015. p. 730–8.

[59] Prabhu Y, Varma M. Fastxml: a fast, accurate and stable tree-classifier for extreme multi-label learning. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; 2014. p. 263–72.

[60] Almagro M, Martinez R, Fresno V, Montalvo S. Icd-10 coding of spanish electronic discharge summaries: an extreme classification problem. IEEE Access 2020;8: 100073–83.

[61] Hasan KS, Ng V. 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; 2014. p. 1262–73. Long Papers.

[62] Gagliardi I, Artese MT. Semantic unsupervised automatic keyphrases extraction by integrating word embedding with clustering methods. Multimodal Technologies and Interaction 2020;4:30.

[63] Blanco A, Viñaspre O Perez-de, Pérez A, Casillas A. Boosting icd multi-label classification of health records with contextual embeddings and label-granularity. Computer Methods and Programs in Biomedicine 2020;188:105264.