Segura Bedmar, I., Colon Ruiz, C., Tejedor Alonso, M.A, Moro Moro, M. (2018). Predicting of anaphylaxis in big data EMR by exploring machine learning approaches, *Journal of Biomedical Informatics*, 87, pp. 50-59.

DOI: 10.1016/j.jbi.2018.09.012

Predicting of anaphylaxis in big data EMR by exploring machine learning approaches

Isabel Segura-Bedmar*[a], Cristobal Colón-Ruíz[a], Miguél Ángel Tejedor-Alonso[b,c], Mar Moro-Moro[b]

[a] *Computer Science Department, University Carlos III of Madrid, Avenida de la Universidad 30, 28911, Leganés, Madrid, Spain*
[b] *Allergy Unit, Hospital Universitario Fundación, Avenida Budapest 1, 28922 Alcorcón, Madrid, Spain*
[c] *Medicine and Surgery Department, Universidad Rey Juan Carlos, 28922 Alcorcón, Madrid, Spain*

Abstract

Anaphylaxis is a life-threatening allergic reaction that occurs suddenly after contact with an allergen. Epidemiological studies about anaphylaxis are very important in planning and evaluating new strategies that prevent this reaction, but also in providing a guide to the treatment of patients who have just suffered an anaphylactic reaction. Electronic Medical Records (EMR) are one of the most effective and richest sources for the epidemiology of anaphylaxis, because they provide a low-cost way of accessing rich longitudinal data on large populations. However, a negative aspect is that researchers have to manually review a huge amount of information, which is a very costly and highly time consuming task. Therefore, our goal is to explore different machine learning techniques to process Big Data EMR, lessening the needed efforts for performing epidemiological studies about anaphylaxis. In particular, we aim to study the incidence of anaphylaxis by the automatic classification of EMR. To do this, we employ the most widely used and efficient classifiers in text classification and compare different document representations, which range from well-known methods such as Bag Of Words (BoW) to more recent ones based on word embedding models, such as a simple average of word embeddings or a bag of centroids of word

*Corresponding author
Email address: isegura@inf.uc3m.es (Isabel Segura-Bedmar*)

embeddings. Because the identification of anaphylaxis cases in EMR is a class-imbalanced problem (less than 1% describe anaphylaxis cases), we employ a novel undersampling technique based on clustering to balance our dataset. In addition to classical machine learning algorithms, we also use a Convolutional Neural Network (CNN) to classify our dataset. In general, experiments show that the most classifiers and representations are effective (F1 above 90%). Logistic Regression, Linear SVM, Multilayer Perceptron and Random Forest achieve an F1 around 95%, however linear methods have considerably lower training times. CNN provides slightly better performance (F1=95.6%).

Keywords: Machine Learning, EMR classification, Bag of centroids, Anaphylaxis, Balancing strategies

---

1. Introduction

Anaphylaxis is a life-threatening allergic reaction that occurs suddenly after contact with an allergen [1]. Despite its possible severe symptoms, even today there is no agreement about clinical criteria for diagnosing anaphylaxis. This lack of specific criteria can result in a failure to properly treat anaphylaxis, with fatal consequences for the patient, because of its early onset and very rapid progression. This has also translated into increased research into the epidemiology of this disorder [1], because epidemiological studies about anaphylaxis are very important in planning and evaluating new strategies that prevent this reaction, but also in providing a guide to the treatment of patients who have just suffered it.

Electronic Medical Records (EMR) are one of the most effective and richest sources for the epidemiology of anaphylaxis, because they provide a low-cost way of accessing rich longitudinal data on large populations. However, researchers have to manually review a huge amount of information, which is a very costly and highly time consuming task. Thus, our goal is to alleviate this burden to researchers, by providing them a system capable to automatically classify if a record describes a case of anaphylaxis or not. This automatic classification

allows researchers to know, among other things, the incidence (total number of new cases) of anaphylaxis among a population.

Although the classification of EMR can be addressed by using a set of rules written by experts, this approach has several drawbacks. Firstly, its coverage is very low because of the richness of the human language with a wide variety of possible expressions to refer to the same object [2, 3]. Moreover, when the number of rules is large, some rules can conflict with one another, hindering their maintenance. No less important is the fact that the rules cannot be reused when the classification problem changes. Unlike the rules-based systems, machine learning algorithms are domain-independent with high predictive performance [4]. In this work, we explore the most widely used and efficient machine learning classifiers in text classification to identify highly probable anaphylaxis cases in EMR. In addition to using the popular bag-of-words approach to represent the records, we use a novel method to represent them using a bag of centroids, which are calculated using the word embeddings from the records. One of the main advantages of our approach, which does not require any knowledge about anaphylaxis, is that it could easily be adapted to identify other diseases and disorders.

Several epidemiological studies put the incidence of anaphylaxis between 50 and 112 episodes per 100,000 person-years [5]. As results, the identification of anaphylaxis cases is a very class-imbalanced problem, which occurs when one class has many more training examples than the other class. This can negatively affect the performance of machine learning techniques, because they tend to be biased towards the majority class, but degrading their performance on minority class. Thus, it is necessary to handle this imbalance issue in order to improve the classification performance of the minority class. The most common approach to overcome this problem is to balance the data distribution on the dataset by using undersampling (decreasing the number of majority classes) and/or oversampling (cloning the minority class instances) methods. Several works [6, 7] have proven that undersampling produces better results than oversampling, which tends to increase the chances of overfitting. On the other hand, representative instances

of the majority class could be ignored by using undersampling.

An innovative solution to this drawback has been recently proposed in [8], where a clustering technique, in particular, the k-means clustering algorithm [9] is used to replace similar instances by their cluster centroid (this is explained with more detailed in Section 3). The authors validated their approach on several datasets from the areas of bioinformatics and quantum physics[1] and a dataset of X-ray images of the breast [2]. However, none of them was composed of texts. Therefore, this work is the first to validate the effectiveness of this undersampling method on an unbalanced dataset of texts.

To sum up, the main contributions of our work are as follows:

- According to the analysis of related work (see Section 2), we propose a bag-of-centroids approach to represent the records, which has never used in the clinical domain. We also explore and compare other ways to represent clinical records such as the BoW model, BoW with tf-idf and the average of word embeddings of all the words in a document.

- We apply a novel undersampling technique [8] based on the use of k-means clustering algorithm. This technique has never been applied before to balance text datasets.

- We provide a detailed analysis of different machine learning algorithms to classify EMR and find the optimum solution for the identification of anaphylaxis cases, with the final goal of reducing the heavy workload in epidemiological studies.

- Our approach does not require the use of domain knowledge, and thereby, can be easily extended to other text classification problems in the medical domain and languages other than Spanish (our dataset is composed of EMR written in Spanish).

[1] http://www.kdd.org/kdd-cup/view/kdd-cup-2004
[2] http://www.kdd.org/kdd-cup/view/kdd-cup-2008

The organization of this paper is as follows. In next section, we discuss previous works in EMR text classification. Section 3 presents the research methodology, including the description of the dataset and the representation of records as instances as well as the balancing technique and machine learning classifiers applied to identify anaphylaxis cases in EMR. In Section 4, we present and discuss the experimental results. Finally, conclusions and potential future work items are identified in Section 6.

2. Related work

The purpose of this section is to discuss the main works that benefit of using Natural Language Processing (NLP) in clinical domain, specifically, for classification of EMR. At the end of this section, we also briefly review the main data balancing techniques.

2.1. Classification of EMR using machine learning algorithms

In text classification, documents are represented by vectors of features, which are the input for machine learning classifiers. Many systems use the popular and simple BoW approach, where tokens (except most common words) are considered as features and are represented by their relative frequency. Other systems exploit NLP tools (such as PoS taggers, noun phrase chunkers or named entity taggers, among others) to obtain linguistic and semantic features to represent texts. Bellow we present a summary of recent work in EMR classification by using NLP and machine learning.

Machine learning and NLP methods were combined in [10] to identify EMR reporting a clinically important brain injury. The authors created an artificial dataset of 3,621 EMR that describe normal and abnormal head computed tomography (CT) scans. The dataset was manually reviewed by doctors to identify those records that report a clinically important brain injury. Texts were tokenized and punctuation, stop-words and superfluous words were removed. Then, texts are represented with vectors of word frequencies using a

BoW model. Three different classifiers were proposed due to its ability to determine the probability of its classifications: K-Nearest Neighbours algorithm (k-NN), Decision Tree classifier and Support Vector Machine (SVM), which obtained the best results (recall of 93.33% and precision of 50%).

EMR were used to classify patient alcohol use in [11]. The records were represented using a BoW approach, which captures the relative frequency of unigrams and bigrams in each document. The system used SVM classifier and was evaluated on a dataset of 2000 records manually classified, providing a very high performance for the detection of current drinkers (F1=89%).

The BoW model was also used in [12] to represent EMR (written in Georgian language). However, in this work, instead of having a binary classification problem, the records had to be classified in three different categories: ultrasonography, endoscopy, and X-ray. SVM and k-NN algorithms were applied, showing very close performance (F1 ranges from 82% to 88%, depending of the category).

Although the BoW model is very effective to represent texts, several works have used other features to represent EMR. Liu et al. [13] adapted the original cTAKES [14] smoking module by adding additional keywords (such as "anxiety" or "dependence") and some sentences (for example "raspy smokers laugh", "smells of cigarette smoke"). Then, these keywords and sentences were used to train a SVM classifier with a radial basis kernel. The approach was trained and tested using a dataset of 400 clinical notes, providing a precision of 94% and a recall of 94%.

cTAKES was also used in [15] to process colonoscopy pathology reports in order to obtain their tokens, part of speech tags and negated terms. These features, along with a dictionary of medical and lay terms, were used to represent each report. A Conditional Random Forest (CRF) was trained to classify pathology reports as either derived from surveillance or non-surveillance colonoscopy in patients with inflammatory bowel disease (IBD). A dataset of 575 reports was manually classified by a gastroenterologist. The system had a precision of 80% and a recall of 77%.

Xu et al. [16] developed a system to identify patients with colorectal cancer (CRC) from EMR. Firstly, the authors defined a list of concepts for CRC using the UMLS methatesaurus [17] and MedLEE [18], a concept extraction system, to detect those concepts in the medical records. MedLEE is able to extract contextual information around a concept, which was used to rule out negated or hypothetical CRC concepts. The authors also trained a SVM classifier to automatically distinguish between positive and negative CRC concepts. The feature set consisted of words and bigrams from a window around each concept, as well as the distance and direction of those words. The feature set also included frequencies of ICD-9 codes [19], Current Procedural Terminology (CPT)[20] codes and normalized frequencies of CRC concepts. For CRC cases detection, the authors tested four different algorithms such as Random Forest (RF), Ripper, SVM and Logistic Regression (LR). The authors created and manually classified a dataset of the medical records (69,820) from 150 patients (with 121 CRC cases). The best results were obtained by Random Forest with a precision of 97% and a recall of 92%.

The creation of gold-standard corpora for machine learning is a very laborious and expensive procedure. To overcome this bottleneck, Ni et al. [21] used active learning and distant supervision to automatically identify sections in EMR. The main idea behind distant supervision is the automatic annotation of texts using knowledge databases, which later are used to train a machine learning classifier [22]. The annotation time and cost were reduced to half by using active learning. Moreover, the distant supervision approach achieved a very good performance (F1=91.2%).

Word embedding models have successfully demonstrated their abilities to capture syntactic and semantic information [23, 24], becoming an interesting alternative to classical approaches for text representation. Word embeddings (word vectors) map words, from a large collections of texts, into a continuous vector space capable of catching main syntactic and semantic relationships between them. Based on distributional hypothesis [25], similar words will have similar vectors because they occur in similar contexts. The recent emergence

of word embeddings through neural networks [26, 27] has provided a novel and promising approach to represent instances in many NLP tasks such as parsing, sentiment analysis, relation extraction, named entity recognition, with tremendous success [28, 29, 30, 31]. However, their use for classification of EMR has rarely been explored. One of the few works is given by Lauren et al. [32]. Each document is represented by averaging the word vectors of its words. Several classifiers, such as Extreme Learning Machine (ELM), SVM and Multilayer Perceptron (MLP), were used to classify EMR based on surgical operation codes. The three algorithms provided very close results. F1 ranges between 91% and 97% for the codes with more instances, and 56%-77% for those with fewer instances.

Many of the recent advances in NLP have been powered to Deep Learning. Text classification is one of the NLP tasks that has greatly benefited from the application of these models such as Convolutional Neural Network (CNN) [33] or Recurrent Neural Network (RNN) [34]. Baket et al. [35] implemented two systems based on SVM: the first one using BoW features and the second one using a set of NLP features such as lemmas, bigrams, named entities, MeSH terms, verb classes and a dictionary of chemical terms. Moreover, the authors trained a CNN model using different pre-trained word embeddings models. The best performance was obtained by the CNN model (F1=81%). SVM achieved an F1 of 76.8% using NLP features and 69.2% with the BoW features.

In this section, we have reviewed several recent works with the goal of classifying EMR. Most works use bag-of word model or linguistic and semantic information (such as unigrams, bigrams, PoS tags or terms from dictionaries) as feature set. In spite of recent growing popularity of word embeddings, they have hardly ever been used to represent EMR for classification. SVM seems to be the most used algorithm classifier with very high performance. It is not possible to draw certain conclusions about the best approach for EMR classification because these works are not comparable among, since they deal with different problems and use different datasets. Although most issues addressed have an unbalanced nature, no work on EMR classification has studied the use of balancing techniques to deal with it.

2.2. Data balancing techniques

Nowadays, machine learning has become a successful technology to solve real complex problems in a wide variety of fields. Many of these problems involve daunting challenges that must be addressed. The unbalanced nature of data is one of these major issues to solve.

The class imbalance problem occurs when one class has many more training examples than the other classes. This negatively affects the performance of machine learning techniques, because they tend to be biased towards the majority class, but degrading their performance on minority class.

The machine learning community mainly recognizes three approaches to deal with the class imbalance problem [36, 37, 38, 39]:(1) feature selection, (2) algorithms and (3) sampling. The class imbalance problem usually implies a high-dimensional feature space to represent the data [40]. Several efforts [41, 38] have been made to show that features selection, a common process in machine learning whose goal is to remove irrelevant and redundant features, can be an effective method to handle the class imbalance problem. Another possible approach to solve the issue of unbalanced data consists of adapting algorithms to the characteristics of the imbalanced datasets. Some examples are cost-sensitive learning [42], one-class classifiers [43] and classifier ensembles [39]. Sampling consists of creating more training examples of the minority class (over-sampling) or removing training examples of the majority class (under-sampling). Synthetic Minority Oversampling Technique (SMOTE) [7] and Adaptive Synthetic (ADASYN) are some of the most popular techniques.

3. Methods

3.1. Dataset

Our dataset consists of computerized records of patients attended at the Emergency Department of the Hospital Universitario Fundación Alcorcón (HUFA) (a major general hospital in Alcorcón, a city in the Madrid metropolitan area of Spain, with a population of 167,354) and also from the diagnosis database of

its Allergy Unit, where all cases of anaphylaxis and other allergic reactions are stored. The records are written in Spanish.

In this work, we use this collection of records as a gold-standard dataset because these records were manually reviewed by two researchers from the Allergy Unit in order to perform epidemiological studies on the incidence of anaphylasis in HUFA [44, 45]. The agreement rate between them was checked with Interobserver Agreement Kappa test [46]. The rate of agreement was 95.08%, with a Kappa index 0.90. In case of not agreement, the final classification of this case was established by consensus. The total number of records is 219,902 (218,079 from the Emergency Department and 1,823 from the Allergy Unit). Only less than one percent of records (2,012) describe anaphylaxis cases. Therefore, it is a very unbalanced dataset.

Spain has a very consistent data protection legislation to the EU Data Protection Directive 95/46 EC. Any transfer or exchange of patient information should comply with the Law 15/1999 on data protection and Law 14/2002 on basic regulation of patient autonomy and rights and obligations of information and clinical data. In particular, Article 16 of Law 4/2002 requires that the access to medical records for judicial, epidemiological, public health, investigation/research or education/teaching purposes is carried out preserving the personal identification data of the patient and separating the personal data from those of clinical character.

In 2016, the Universidad Carlos III de Madrid (UC3M) and HUFA signed an agreement where both organizations expressed their interest in joint cooperation towards researching about new technologies for supporting epidemiological studies and committed to complying with applicable data protection laws. Once the agreement was signed, HUFA provided us the clinical records without any personal data of patients. In particular, the following structured fields were removed: names of patients, their health proxies, information from family members, names of doctors, identification numbers, telephones and faxes, geographic locations, and dates. To even more preserve patient anonymity, we also processed the records with regular expressions to remove telephone numbers,

dates and people names from the unstructured fields.

Training and validation datasets are used for training machine learning methods and optimizing their parameters. Moreover, a test dataset is needed to obtain honest assessments of the performance of the predictive models. We randomly split the whole dataset into training and test subsets. The split ratio was 75-25% (see Table 1). The datasets were randomly generated and the only consideration made was that the ratio between positive and negative examples should be equals to the ratio in the whole dataset. In other words, each subset only contains 0.9% of positives examples (records describing anaphylaxis cases).

Dataset	**#negatives**	**#positives**
Training	163,417 (99.1%)	1,509 (0.9%)
Test	54,473 (99.1%)	503 (0.9%)
Total	217,890 (99.1%)0	2,012 (0.9%)

Table 1: Datasets for anaphylaxis classification.

3.2. Feature set

One of the main goals of this work is to evaluate and compare the impact of different document representations on the performance of several classifiers to detect EMR describing anaphylaxis cases.

3.2.1. Bag-of-Words approach

First, we consider as baseline the BoW approach for document representation. This model allows us to represent documents as vectors, where each position in the vector represents a specific word, and the value at that position denotes the frequency of that word in the document.

3.2.2. Bag-of-Words with tf-idf approach

We extend the BoW baseline by weighting the BoW vectors with tf-idf (frequency-inverse document frequency) [47], instead of using the frequency of the words. This allows us to measure the word relevance in a collection of texts.

Averaging of word embeddings

Like in [32], we also represent each of our texts by averaging the word embeddings of the words occurring in it. To do this, we explore two word embedding models. The first model is the Cardellino's word embedding model [48], a pre-trained model created from several Spanish collection texts such as Spanish Wikipedia (2015), the OPUS corpora [49] or the Ancora corpus [50], among others. It contains a total of 1,000,653 words and the dimension of its word vectors is 300.

Moreover, we created our own word embedding model using the Word2vec model [26] on the whole collection of EMR, which HUFA provided us (see Table 1). Before training the word embedding model, we cleaned and processed the records, which were split into sentences using the NLTK library[3] (it includes support for Spanish). Accented letters were replaced by non-accented equivalents and then non-alphanumeric characters and stopwords were removed. Finally, tokens were stemmed by using the Snowball implementation of the Porter's word stemming algorithm [51] for Spanish. The goal of stemming is to replace all the possible inflected forms of a word to an only common base form (stem). Finally, we used the gensim library[4] to train our word embedding model. We used the Skipgram model with a window of size 10 and ignored the words with a frequency lower than 5. The skipgram model is better when the training data is limited [26], as is our current dataset. Moreover, following previous works [48, 52], we set the dimension of word embeddings to 300, because this dimension often obtains better results than others [53]. The trained model had a vocabulary size of 38,274 stems. It should be noted that its size is very much smaller than the vocabulary of the Cardellino's word embedding model, with more than 1 million of words.

[3] http://www.nltk.org/

[4] https://radimrehurek.com/gensim/

Bag-of-Centroids approach

Moreover, we propose a fourth method based on a bag-of-centroids approach, similar to the bag-of-words idea. Clustering the word embeddings allows us to obtain groups of word vectors that are relatively close to one another in the space (hence, they represent similar words) and relatively far from those in other clusters. The idea is to represent the clinical records using these clusters, instead of using directly their word embeddings as in the previous approach [32].

We applied the k-means algorithm [54] to cluster the word embeddings due to its efficiency and simplicity, though other methods could be applied. The only parameter required by k-means is the number of clusters, which is directly related with the size of their clusters, that is, with the number of word vectors in each cluster. This size should not be too large (many words in the cluster may not be similar) or too small (similar words would probably be left out of the cluster). We set the size to 10 words and then used it to divide the vocabulary size of the model. For the word embedding model trained on EMR, the value of K equals to 3,828, while its value is far greater for the Cardellino's word embedding model, with a total of 100,065 clusters. We also tried with a size of 5 words, but we finally has to discard it, because the document representation using the bag-of-centroids approach is too demanding computationally, specially for the Cardellino's word embedding model. In addition to the clusters, the k-means algorithms also defines the centroid of each cluster, as the mean value of the vectors within it. These k centroids form our bag of centroids.

Now we describe in detail the process to represent a document with the bag-of-centroids approach:

- Each record must be cleaned and preprocessed following the same steps in the creation of each word embedding model. Thus, when we use the bag-of-centroids built from the Cardellino's word embedding model, words are not stemmed.

- Then, we iteratively looked for each word (or stem if we are using the word embedding trained on EMR) into the corresponding word embedding

model to retrieve its vector. If the word (or stem) is not found, it is ignored.

- For each found word (stem), we obtained the cluster to which its word vector belongs, increasing its count by one.
- Thus, the final result is a vector of dimension K, where each element represents to one of the clusters. The corresponding value of a element is the number of words (stems) in the record belonging to this cluster.

3.3. Clustering-based undersampling technique

Because the identification of anaphylaxis cases is a unbalanced problem class, We used a clustering-based method to undersample the majority class. One of the main disadvantages of undersampling techniques is that representative instances of the majority class could be ignored. An innovative solution to this drawback has been recently proposed in [8], where a clustering technique, in particular, the k-means clustering algorithm is used to represent similar instances of the majority class in the training dataset. Lin et al. [8] already validated this approach on several datasets from the areas of bioinformatics and quantum physics[5] and a dataset of X-ray images of the breast [6]. However, none of them was composed of texts. Thus, our work is the first to apply this methods on an unbalanced dataset of texts, in particular, EMR.

In particular, this approach obtains as many clusters in the majority class as the number of training instances in the minority class. Then, all the instances in a cluster are replaced by its cluster centroid (center), thus achieving to reduce the number of instances in the majority class to the number of instances in the minority class. A cluster centroid is the mean value of all the instances (which were already represented as vectors using a bag-of-centroids approach) in a given cluster. This means that some centroids may not be real instances.

In this work, we propose an alternative to avoid non-real instances. For each cluster using the Euclidean distance, we calculate the nearest neighbour

[5] http://www.kdd.org/kdd-cup/view/kdd-cup-2004
[6] http://www.kdd.org/kdd-cup/view/kdd-cup-2008

to its centroid. Then, the examples in that cluster are replaced by this nearest neighbour, instead of by the centroid. In this way, we do not only achieve that both classes are balanced, but also the majority class only contains real instances.

3.4. Classifiers

In this paper, we compare several classifiers to assess their performance for the detection of anaphylaxis cases: Multinomial Naive Bayes [55], Support Vector Machine (SVM)[56], Logistic Regression [57], k-Nearest Neighbours algorithm (k-NN) [58], Multilayer Perceptron (MLP) [59] and Random Forest [60]. We also use a majority voting classifier of all classifiers. These classifiers were chosen because of their large use and good performance in text classification. We also propose to compare the classical machine learning algorithms to a Convolutional Neural Networks (CNN) [61], because it has been successfully used in text classification.

Classical machine learning algorithms

Multinomial Naive Bayes classifier has been proven very effective for text classification. It is a probabilistic model based on theorem of Bayes. This classifier calculates the probabilities of each text belonging to each class and then selects the class with the maximum probability. The adjective naive comes from the assumption that all features are independent given class. Although such an independence assumption is not usually true, the algorithm often performs surprisingly well with a fast computational time. Moreover, it requires a small amount of training data, is very easy to implement and is also very scalable. Despite its simplicity, the Naive Bayesian classifier often exceeds more sophisticated classification algorithms.

SVM, perhaps one of the most popular and successful classifiers, is a non-probabilistic linear classifier that tries to find the hyperplane that best separates the classes, maximizing the margin between them while, at the same time, minimizing the number of misclassification errors. The main reason of its success

is that most text classification problems are linearly separable [62]. Moreover, SVM is able to learn, irrespective of the dimensionality of the feature space, because it is based on maximization of the margin, not the number of features [62]. If the classes are separable by a wide margin, then the model will be able to generalize even with a very large number of features. Another advantage of SVM is that it also works well classifying sparse instances [63], as the ones generated by the bag-of-centroids approach. In SVM, we can user several kernel functions such as linear kernel, polynomial kernel, sigmoid kernel or radial basis function (RBF) kernel. A kernel function transforms the input space into a high dimensional space where the problem can be represented as a linear problem. Linear kernel is much faster, while RBF generally provides better performance. However, when the number of features is large, which is typical in text classification, the RBF kernel does not provide better performance than using the linear kernel. In our experiments, we compare linear and RBF kernels.

Logistic Regression is a linear classifier, which can be used to predict the probability of an event. Its main advantage is that its results have an easier interpretation than those obtained by other classification algorithms. Moreover, this algorithm provides a regularization parameter to avoid over-fitting. Among their disadvantages, it requires much more data than other classifiers to obtain stable and accurate results. Moreover, it is not able to capture complex relationships in the data.

k-NN is one of the simplest classification algorithm. It is based on the idea that the closer instances are, the more probability they belong to the same class. In this way, one of its main advantages is that it is a lazy classifier because it does not create a training model from the training dataset, but rather compares the test instance with all instances to determine its class. Moreover, the classifier does not depend on the data distribution.

We also use a Multilayer Perceptron (MLP), a class of feedforward artificial neural network with error back propagation learning algorithm. MLP is able to generalize to non-linear separable data, thanks to its neurons in hidden layers use a nonlinear activation function. Conversely, however, its multilayered structure

requires substantially longer training time for learning than other algorithms. Moreover, it is needed to tune the hyperparameters of the network (for example, number of layers and neurons in each layer, number of iterations). This is clearly a very costly process, since these hyperparameters are usually determined by trial-and-error in order to find the simplest structure that obtains acceptable results. We use the ReLU [64] ($f(x) = max(0, x)$) activation function, due to its superior effectiveness and efficiency in contrast to other more complex functions, such as sigmoidal or logistic functions. We empirically set the number of hidden neurons to n = 100.

Random forest is an ensemble classifier of a collection of decision trees by randomly selecting examples from the training data. The final prediction is calculated by aggregating the predictions of each tree. Learning from different trees leads to mitigate the over-fitting as well as errors due to bias and variance in the decision trees. Random forests are more robust and generally exhibit better results than decision trees. Regarding the most appropriate number of trees in a forest, several studies [65] have observed that a larger number of trees implies a significant increase in computational costs, without an improvement in results. Its optimal values ranges between 64 and 128 trees [65]. We set the number of trees to n = 100.

Convolutional Neural Network

During the 2010s, deep learning methods are overshadowing the traditional machine learning algorithms in many NLP applications [66, 67, 68, 69, 70]. In addition to the high performance of deep learning methods, they do not rely on any hand-engineered features, but rather are able to learn the most appropriate features for a given task. Therefore, deep learning is probably the leading and most successful approach for NLP currently. Convolutional Neural Network (CNN) [61] and Recurrent Neural Network (RNN) [71] are the most commonly used deep learning architectures.

From a theoretical point of view, RNN, which has a sequential architecture, should be more appropriate for sequence modelling tasks (such as machine

translation, language modelling or speech recognition), because these tasks need to represent complicated context dependencies. Likewise, the hierarchical architecture of CNN should be more useful for text classification, where the detection of representative patterns can be the key to solve the problem. In fact, CNN architectures have proved to outperform the state-of-the-art algorithms in text classification [72, 33, 73], because they are able to extract the most informative ngrams describing a text.

We propose a very simple CNN architecture including one input layer, one convolution layer, one max-pooling layer and an output layer, which is a logistic regression algorithm. In particular, we used a softmax classifier. A more detailed description of this architecture can be found in [72].

Records were represented by using word embeddings of their content words. We performed experiments using the two word embedding models used in this work: Cardellino's word embedding model and our EMR word embedding model. We cleaned records by removing stopwords and all non alphanumeric characters. Words were also stemmed if we use the EMR word embedding model. Both models have a dimension of 300. In the input layer, each neuron corresponds to one word, which is represented by its word embedding.

To know the most appropriate number of neurons to represent our collection of records, we calculated the average number of words per record, which is 238.4 with a standard deviation of 303.2. Moreover, we observed that 99.9% of records had less than 2,000 words. For this reason, we set the number of neurons to 2,000 for representing each record. In this way, we only considered the 2,000 right-most tokens of each record and ignored the remaining ones. To take this decision, we studied a set of records and observed that, in general, their ending parts contained more discriminative information for the diagnosis of anaphylaxis than the beginning parts. Therefore, we applied truncation and zero-padding to handle records with different length than 2,000 tokens. If a record is too long, its left-most part is truncated. Likewise, if a record has a length smaller than 2,000, its left-most part is extended by zero-padding. In our experiments, we kept the word embeddings inputs fixed during the training of the network.

In the convolutional layer, filters are applied to the input matrix. A filter is a sub-matrix which slides over the input matrix. Several filters can be used in this layer and their maps are stacked to produce its output. We experimentally set the number of filters to 64 and a filter size of [2,3,5]. We used Rectified Linear Unit (ReLU) as activation function.

The pooling layer tries to obtain more compact representations, preserving relevant features while removing irrelevant details. This layer can be performed in several ways, for example, calculating the average, taking the maximum, or as a linear combination of its inputs. We applied max-pooling, which gets the maximum value for each convolution. Finally, the outputs from the pooling layer are concatenated into a single vector, which can be considered as the text representation learned by the CNN for the input record. This vector is passed to a softmax classifier layer to predict if the record is a positive instance (describes a case of anaphylaxis) or not.

4. Results

Because accuracy is not a meaningful measure when the dataset is unbalanced, we use precision, recall and F1 to compare the techniques studied in this work. Precision represents the number of documents (records) correctly classified divided by the total number of documents classified as anaphylaxis cases by our system. Recall is the ratio between the number of documents correctly classified divided by the total number of documents that truly describe an anaphylaxis case (that is, the size of the minority class). F1 metric is a weighted harmonic mean of recall and precision [74], which tries to balance both metrics.

All the experiments were conducted in Python using scikit-learn library for classification. We used GridSearchCV, a class from scikit-learn library for searching the best parameters for a specific classifier, to conduct 10-fold cross validation on the different classifiers.

4.1. Results using BoW and TF-IDF

As baseline, we represented the records using the standard BoW representation as well as the TF-IDF vector space model. We applied the machine learning classifiers described above. Moreover, we generated a majority voting classifier ensemble of these classifiers to check if it provides an improvement of the results.

Classifier	**P**	**R**	**F1**	**Training time (s)**
MultinomialNB	0.776	0.916	0.840	0.164
LogReg	0.967	**0.932**	0.949	17.37
Linear SVM	0.966	**0.930**	0.948	2.919
SVM (RBF kernel)	**1**	0.318	0.482	562.7
k-NN	**1**	0.862	0.926	**0.057**
MLP	0.955	**0.930**	0.942	2464.8
Random Forest	**1**	0.878	0.935	52.50
Voting	0.995	0.908	**0.950**	2449.9

Table 2: Baseline results using BoW model. The lowest training time and top scores are bold.

Table 2 shows the baseline results using the BoW approach to represent the records. The training time of each algorithm is also shown. The majority voting classifier achieves the highest F1 (95%) and a precision of 99.5%. Logistic Regression and Linear SVM also achieve the top F1. Linear SVM requires significantly less training time than the other top classifiers. Indeed, Linear SVM is more than 816 times faster than the voting classifier. MLP also achieves very close performance (F1=94.2%), however this classifier is very costly in terms of training time.

The fastest classifier (with a training time of 0.057 s) is k-NN, which gives a very close score (92.6%) to the highest F1. Multinomial Naive Bayes also requires a very low training time (0.164 s), however its F1 (84%) is 11 points lower than the top.

SVM (RBF kernel), k-NN and Random Forest achieve the highest precision

(100%). While k-NN and Random Forest retain an acceptable recall of more than 87.8%, RBF-SVM leads to a low value of recall (31.8%). In terms of F1, the baseline results show that the detection of anaphylaxis cases in EMR can be successfully represented as a binary linear SVM problem. k-NN can be also a good choice in terms of training time, providing a high F1 of 92.6%, only two points less than the top F1.

Table 3 shows the classifiers' results and their training times using the tf-idf space vector model to represent the records.

Classifier	**P**	**R**	**F1**	**Training time (s)**
MultinomialNB	**1**	0.145	0.253	0.173
LogReg	0.991	0.910	0.949	5.759
Linear SVM	0.989	0.929	**0.953**	1.642
RBF-SVM	0	0	0	429.1
k-NN	**1**	0.747	0.855	**0.298**
MLP	0.967	**0.932**	0.949	4001.9
Random Forest	**1**	0.862	0.926	58.53
Voting	0.993	0.908	0.949	3767.6

Table 3: Results using tf-idf space vector model. The lowest training time and top scores are bold.

The BoW approach with tf-idf obtains the same top F1 (95%) than the standard BoW approach. The use of tf-idf space vector model seems to slightly improve the precision for the classifiers: Logistic Regression, Linear SVM and MLP. Their training times are also lower than using the standard BoW approach. MLP also achieves the top F1 (94.9%), however it requires more than one hour to train its model compared to only 1.6 seconds needed by Linear SVM. Not even the majority voting classifier can be considered a better alternative to Linear SVM, because it has a very high training time.

Multinomial Naive Bayes (NB) also achieves a significant improvement of its precision, reaching 100%, but with a strong negative effect on its recall (14%),

and thereby, also on its F1 (25%). In general, the tf-idf model affects negatively the recall rates of the classifiers. While for some classifiers (Random Forest, Logistic Regression, Linear SVM and MLP) the negative effect is moderated (with a loss of less than 2 points), the other classifiers, Multinomial NB and non linear SVM, suffer more severe drops in their recall levels. It is difficult to know why the representation based on TF-IDF had a very negative effect on the recall of these classifiers, compared to the recall obtained using the BoW approach. The method of TF-IDF smooths the high term frequencies. This fact may be negatively affecting the probability of the scarce positive instances in the probabilistic model based on Multinomial NB. In the case of kernel-based SVM, the use of TF-IDF, instead of using term frequency, could be substantially smoothing the relevant terms in positive instances. This fact seems to make difficult finding the optimal hyperplane that separates positive and negative instances.

The high performance obtained by the linear classifiers may suggest that the representation of records using tf-idf vectors for the identification of anaphylaxis cases seems to have a linear nature. As a conclusion, after comparing BoW and TF-IDF representations, the TF-IDF model does not improve the performance of classifiers.

4.2. Results using the average of word embeddings

Classifier	P	R	F1	Training time (s)
MultinomialNB	0	0	0	0.5611
LogReg	0.973	0.669	0.786	13.74
Linear SVM	0.978	0.906	0.941	**4.236**
RBF-SVM	0	0	0	260.0
k-NN	0.974	**0.914**	0.943	8.867
MLP	0.970	**0.912**	0.940	204.9
Random Forest	**0.995**	0.900	**0.945**	364.5
Voting	**0.995**	0.900	**0.945**	380

Table 4: Results using the average of word embeddings from EMR. The lowest training time and top scores are bold.

The next step in our experimentation was to assess the ability of word embeddings to represent EMR. Our goal is to determine if word embeddings are able to capture the sufficient information for the classification of EMR describing anaphylaxis cases. As it was described in the previous section, we use a very simple approach to represent a document as the average of its word embeddings of the content words occurring in the document. We experiment with two different word embedding models: the first model was trained using the whole collection of EMR provided by HUFA and the other was the Cardellino's word embedding model [48].

The best classifiers using the word embedding model trained on the collection of EMR are Random Forest and the majority voting ensemble (F1=95%), with a precision of 100%. However, both methods take a significant amount of training time. On the other hand, Linear SVM or k-NN can train their models within a short training time (less than 10 seconds) and very competitive performance (F1=94%).

The document representation based on the averaging of the word embeddings does not provide better performance than the standard BoW approach.

Moreover, Multinomial NB and RBF-SVM are unable to classify any instance. The only possible advantage of this document representation is that the training time for MLP is more than ten times faster than using the BoW model or its extension with tf-idf. This may be due to the fact that neural networks learn faster if they use pre-trained word embeddings as input.

Classifier	**P**	**R**	**F1**	**Training time (s)**
MultinomialNB	0	0	0	0.3875
LogReg	0.967	0.777	0.862	13.74
Linear SVM	0.968	0.904	0.935	**8.48**
RBF-SVM	0	0	0	288.40
k-NN	0.958	0.902	0.930	10.22
MLP	0.936	**0.910**	0.923	143.6
Random Forest	**1**	0.854	0.921	328.87
Voting	0.997	0.898	**0.945**	375.6

Table 5: Results using the average of word embeddings from Cardellino's word embedding model. The lowest training time and top scores are bold.

A similar performance is obtained using the Cardellino's word embedding model. The majority voting classifier provides the top F1 (94%), but with a very high training time (more than 6 minutes). Linear SVM and k-NN are the second top classifiers with a F1 of 93% and very low training times (approximately 10 seconds). Linear SVN shows very close results regardless of the approach used to represent the records (its F1 ranges between 93% and 95%). Likewise, MLP has a similar behaviour, which is not affected by the document representation. Multinomial NB and RBF-SVM are unable to classify any instance, as happened with the word embedding model trained on the collection of EMR. If we do not consider these two classifiers, the lowest performance is achieved by Logistic Regression (F1=86%). Overall, both word embedding models provide very close results and training times. They are not able to overcome the simple BoW model.

4.3. Results using Bag-of-Centroids approach

Classifier	P	R	F1	Training time (s)
MultinomialNB	0.235	**0.936**	0.376	0.332
LogReg	0.937	0.930	0.934	20.31
Linear SVM	0.898	0.928	0.913	15.91
RBF-SVM	**1**	0.898	**0.946**	193.49
k-NN	**1**	0.8986	**0.946**	**0.073**
MLP	0.943	**0.936**	0.940	127.2
Random Forest	**1**	0.892	0.943	79.94
Voting	0.974	0.916	0.944	200

Table 6: Results using bag of clusters from the word embedding model trained on EMR. The lowest training time and top scores are bold.

Classifier	P	R	F1	Training time (s)
MultinomialNB	0.676	0.918	0.779	0.263
LogReg	0.953	0.926	0.939	19.77
Linear SVM	0.939	0.920	0.929	32.73
RBF-SVM	0.994	0.379	0.549	442.56
k-NN	**1**	0.842	0.914	**0.049**
MLP	0.945	**0.936**	0.941	2792.7
Random Forest	**1**	0.854	0.921	85.15
Voting	0.993	0.910	**0.950**	3000

Table 7: Results using bag of clusters from the Cardellino's word embedding model. The lowest training time and top scores are bold.

Tables 6 and 7 show the results using the bag-of-centroids approach. In general, the results are very similar regardless the word embedding model used. However, the Cardellino's word embedding model seems to increase significantly the precision of Multinomial NB, with an improvement of 44%. On the contrary, this same model seems to negatively affect the recall of the radial basis SVM

with a significant decrease of 52%. Surprisingly, this algorithm achieves a high F1 is 94% when we use the EMR word embedding model.

The bag-of-cluster approach also provides very close results, regardless the classifier used. All classifiers are above 90% of F1, except Multinomial Nave Bayes, which only gets an F1 of 38% (using the EMR word embedding model) or 78% (using the Cardellino's word embedding model) and the radial basis SVM (F1=55% when we used the Cardellino's word embedding model). For the EMR word embedding model, five of them (k-NN, Random Forest, RBF-SVM, MLP and voting) achieves the top F1 (94%). Non-linear SVM shows (RBF-SVM) shows slightly better performance than linear SVM (93%), but the non-linear kernel requires considerably more training time than the linear one.

k-NN, Linear SVM and Logistic Regression provide high F1 (91-94%) with very low training times (less than 20 seconds). The fastest algorithm is k-NN with a training time less than one second. This may be due to the fact of k-NN is a lazy classifier. The training times are also very close using both models. The only relevant difference is for MLP. This make sense because the vectors for the bag-of-centroids based on the Cardellino's word embedding model have a larger dimension than using the word embedding model trained on EMR (100,065 clusters versus than 3,828 clusters).

The best performance is obtained with the majority voting classifier trained using the Cardellino's word embedding model (99% precision, 91% recall and an F1 of 95%). The difference may be due to the Cardellino's word embedding model is much larger than the word embedding model trained using EMR. Other classifiers achieve very close performance (Logistic Regression, Linear SVM, MLP, Random Forest).

As happened with the representation using the average of word embeddings, the bag-of-centroids approach provides very close results to those obtained by the BoW model.

4.4. Results using the clustering-based undersampling method

From the previous results, we can conclude that the classification of anaphylaxis cases can be successfully using machine learning classifiers, despite the fact that the dataset is very imbalanced. However, we decided to study the impact of an undersampling technique on the results.

As it was described before, only less than one percent of records (2,012) describe anaphylaxis cases in our dataset, a very small amount compared to the number of negative instances (217,890). To reduce the great imbalance of data, we apply the clustering-based method to undersample the majority class, which was described above. To simplify the experimentation, we only focus on the bag-of-clusters approach because the k-means algorithm was already used to the centroids. In particular, we used the Cardellino's word embedding model because it provides slightly better performance than the other word embedding model.

The results obtained when the clustering-based undersampling method are shown in Table 8. The best performance (F1=95.3%) is achieved by Random Forest with a precision of 99.7%. The classifier with the highest recall is Linear SVM, however its precision has gone drastically down to 63.9% compared to 93.9% of precision when we only use the bag-of-centroids approach, without applying any undersampling technique. Most classifiers (such as Multinomial NB, Logistic Regression, Linear SVM) are negatively affected by the clustering-based undersampling technique.

Classifier	P	R	F1
MultinomialNB	0.379	0.924	0.538
LogReg	0.742	0.962	0.831
Linear SVM	0.639	**0.958**	0.766
RBF-SVM	0.996	0.532	0.694
k-NN	0.993	0.846	0.914
MLP	0.759	0.952	0.844
Random Forest	**0.997**	0.912	**0.953**
Voting	0.865	0.922	0.893

Table 8: Clustering-based undersampling results (with the Cardellino's word embedding model). The top scores are bold.

4.5. Results using CNN

We finally present the results obtained with our CNN model in Table 9. We compare the use of two different word embeddings models: the pre-trained Cardellino's model and the model trained using our collection of EMR. Both CNN models provide very close results to the top ones (F1=95%) obtained by some of the classical machine learning algorithms (Logistic Regression, Linear SVM, MLP and Random Forest).

We implemented the network using Keras [7] and trained the model on GPU (NVidia Titan XP) because GPUs are more usually efficient than CPUs. We train with a batch size of 256 and an initial learning rate of 0.001. To fit network parameters, we used Adam [75] with a base learning rate 0.001. We trained for 50 epochs. Using GPU instead of CPU, the average training time per epoch is 60 seconds (compared to 2759 seconds with CPU). In total, training took 3,000 seconds on a NVidia Titan XP. Although GPU achieves a considerable time reduction compared to CPU (50 minutes versus 38 hours) to train the CNN model with a high performance (F1=95.6%), many of the classical machine learning classifiers obtain similar performance and with much less training time

[7] https://keras.io/

needed.

Word embedding model	P	R	F1
EMR	**0.989**	**0.926**	**0.956**
Cardellino's	0.985	0.922	0.952

Table 9: CNN results.

5. Discussion

There are several settings that are capable to provide very high performance in the prediction of anaphylaxis in clinical records. Below, we summarize the approaches that give the top F1. We also show their training times:

1. CNN trained on the EMR's word embedding model: F=95.6%, 50 minutes for training.
2. CNN trained on the Cardellino's word embedding model: F=95.2%, 50 minutes for training).
3. Majority voting on the bag of clusters from the Cardellino's word embedding model: F=95%, 50 minutes.
4. Logistic Regression on BoW with tf-idf: F=94.9%, 6 seconds.
5. Linear SVM on BoW with tf-idf: F=95.3%, 2 seconds.
6. MLP on BoW with tf-idf: F=94.9%, 66 minutes.
7. Logistic Regression on BoW: F=94.9%, 17 seconds.
8. Linear SVM on BoW: F=94.8%, 3 seconds.
9. Majority voting on BoW: F=95%, 41 minutes.

Overall, the top performance is an F1 of 95.6% provided by CNN with the EMR word embedding model. However, linear classifiers (such as Logistic Regression or Linear SVM) using the simple BoW representation are capable of predicting anaphylaxis cases in EMR with the same high performance than other complex methods, with the advantage of requiring much less training time. Thus, we think that the classification of anaphylaxis cases seems to be a linear

problem and can be efficiently solved by these linear classifiers and the BoW representation. Using deep learning models to solve this problem could be as using a sledgehammer to crack a nut.

Our allergists manually reviewed the set of false positives and negatives, which were produced by one of this best setting (Linear SVM using BOW). Allergists concluded that the false positives correspond to records that include a broad range of signs and symptoms associated with an allergic reaction, but the overall description does not correspond to an anaphylaxis case, because the symptoms described are relatively mild. The false negatives contain a large amount of vocabulary very related and used in the description of anaphylactic reactions. However, none of the signs or symptoms described in them corresponded to serious symptoms such as cardiac arrest, feeling of doom, low blood pressure or edema of glottis.

6. Conclusion

Knowing the epidemiology of life-threatening events such as anaphylaxis is crucial to improve patient safety. Epidemiological studies do not only allow to identify determinants of a specific disease and adopt preventive strategies to reduce its impact, but also provide valuable information to plan resources for the treatment of patients, who already suffer this disease. In summary, these studies can support healthcare professionals in making accurate decisions. However, most epidemiological studies are very costly and time consuming, because of the vast amount of data (clinical records, databases, etc) which must be reviewed. The main goal of this work is to explore different machine learning methods to identify anaphylaxis cases in EMR. The automatic identification of these cases may help to drastically reduce the cost and burden of epidemiological studies on this disease.

In this work, we compared different methods to represent EMR and applied some of the most popular classifiers in order to identify clinical records describing anaphylaxis cases. Our experiments show that the prediction of anaphylaxis

cases is a linear problem that can be efficiently solved by using linear classifiers such as Linear SVM or Logistic Regression and the BoW representation. The undersampling method does not seem to improve the results. The top F1 is achieved by a simple CNN architecture, however it requires around 50 minutes to train the network.

Once we have already achieved an efficient module for the automatic identification of anaphylaxis cases described in EMR, we plan to develop a information extraction system that allows us to detect specific concepts and relationships between them, providing a more comprehensive support to doctors in the performing of epidemiological studies. To do this, we plan to combine different NLP and deep learning methods in order to handle the challenges (such as misspellings, redundancy, ambiguity) in EMR. Moreover, we also plan to apply our classification system to other diseases or disorders as well as EMR written in other languages than Spanish.

Acknowledgments

This work was supported by the Research Program of the Ministry of Economy and Competitiveness - Government of Spain, (DeepEMR project TIN2017-87548-C2-1-R).

References

[1] H. A. Sampson, A. Muñoz-Furlong, R. L. Campbell, N. F. Adkinson, S. A. Bock, A. Branum, S. G. Brown, C. A. Camargo, R. Cydulka, S. J. Galli, et al., Second symposium on the definition and management of anaphylaxis: summary reportsecond national institute of allergy and infectious disease/food allergy and anaphylaxis network symposium, Annals of emergency medicine 47 (4) (2006) 373–380.

[2] I. Segura-Bedmar, P. Martínez, C. de Pablo-Sánchez, A linguistic rule-based approach to extract drug-drug interactions from pharmacological documents, BMC bioinformatics 12 (2) (2011) S1.

[3] Q. L. Nguyen, D. Tikk, U. Leser, Simple tricks for improving pattern-based information extraction from the biomedical literature, Journal of biomedical semantics 1 (1) (2010) 9.

[4] Y. Wang, L. Wang, M. Rastegar-Mojarad, S. Moon, F. Shen, N. Afzal, S. Liu, Y. Zeng, S. Mehrabi, S. Sohn, et al., Clinical information extraction applications: A literature review, Journal of biomedical informatics.

[5] M. Tejedor, M. Moro-Moro, M. Múgica, Epidemiology of anaphylaxis, Clinical & Experimental Allergy 45 (6) (015) 1027–1039.

[6] B. X. Wang, N. Japkowicz, Boosting support vector machines for imbalanced data sets, Knowledge and Information Systems 25 (1) (2010) 1–20.

[7] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, Smote: synthetic minority over-sampling technique, Journal of artificial intelligence research 16 (2002) 321–357.

[8] W.-C. Lin, C.-F. Tsai, Y.-H. Hu, J.-S. Jhang, Clustering-based undersampling in class-imbalanced data, Information Sciences 409 (2017) 17–26.

[9] S. Lloyd, Least squares quantization in pcm, IEEE transactions on information theory 28 (2) (1982) 129–137.

[10] D. A. Szlosek, J. Ferrett, Using machine learning and natural language processing algorithms to automate the evaluation of clinical decision support in electronic medical record systems, eGEMs 4 (3).

[11] L. Lix, S. N. Munakala, A. Singer, Automated classification of alcohol use by text mining of electronic medical records, Online Journal of Public Health Informatics 9 (1).

[12] M. Khachidze, M. Tsintsadze, M. Archuadze, Natural language processing based instrument for classification of free text medical records, BioMed research international 2016.

[13] M. Liu, A. Shah, M. Jiang, N. B. Peterson, Q. Dai, M. C. Aldrich, Q. Chen, E. A. Bowton, H. Liu, J. C. Denny, et al., A study of transportability of an existing smoking status detection module across institutions, in: AMIA Annual Symposium Proceedings, Vol. 2012, American Medical Informatics Association, 2012, p. 577.

[14] G. K. Savova, J. J. Masanz, P. V. Ogren, J. Zheng, S. Sohn, K. C. Kipper-Schuler, C. G. Chute, Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications, Journal of the American Medical Informatics Association 17 (5) (2010) 507–513.

[15] J. K. Hou, M. Chang, T. Nguyen, J. R. Kramer, P. Richardson, S. Sansgiry, L. W. DAvolio, H. B. El-Serag, Automated identification of surveillance colonoscopy in inflammatory bowel disease using natural language processing, Digestive diseases and sciences 58 (4) (2013) 936–941.

[16] H. Xu, Z. Fu, A. Shah, Y. Chen, N. B. Peterson, Q. Chen, S. Mani, M. A. Levy, Q. Dai, J. C. Denny, Extracting and integrating data from entire electronic health records for detecting colorectal cancer cases, in: AMIA Annual Symposium Proceedings, Vol. 2011, American Medical Informatics Association, 2011, p. 1564.

[17] O. Bodenreider, The unified medical language system (umls): integrating biomedical terminology, Nucleic acids research 32 (suppl_1) (2004) D267–D270.

[18] J.-H. Chiang, J.-W. Lin, C.-W. Yang, Automated evaluation of electronic discharge notes to assess quality of care for cardiovascular diseases using medical language extraction and encoding system (medlee), Journal of the American Medical Informatics Association 17 (3) (2010) 245–252.

[19] W. H. Organization, P. M. I. Corporation, ICD-9-CM: International Classification of Diseases, 9th Revision: Clinical Modification, Vol. 1, PMIC (Practice Management Information Corporation), 1998.

[20] A. M. Association, Current procedural terminology: CPT, American Medical Association, 2007.

[21] J. Ni, B. Delaney, R. Florian, Fast model adaptation for automated section classification in electronic medical records., in: MedInfo, 2015, pp. 35–39.

[22] M. Mintz, S. Bills, R. Snow, D. Jurafsky, Distant supervision for relation extraction without labeled data, in: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2, Association for Computational Linguistics, 2009, pp. 1003–1011.

[23] R. Socher, C. C. Lin, C. Manning, A. Y. Ng, Parsing natural scenes and natural language with recursive neural networks, in: Proceedings of the 28th international conference on machine learning (ICML-11), 2011, pp. 129–136.

[24] R. Fu, J. Guo, B. Qin, W. Che, H. Wang, T. Liu, Learning semantic hierarchies via word embeddings, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vol. 1, 2014, pp. 1199–1209.

[25] Z. S. Harris, Distributional structure., Word.

[26] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in neural information processing systems, 2013, pp. 3111–3119.

[27] J. Pennington, R. Socher, C. Manning, Glove: Global vectors for word representation, in: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1532–1543.

[28] J. Turian, L. Ratinov, Y. Bengio, Word representations: a simple and general method for semi-supervised learning, in: Proceedings of the 48th annual meeting of the association for computational linguistics, Association for Computational Linguistics, 2010, pp. 384–394.

[29] R. Socher, J. Bauer, C. D. Manning, A. Y. Ng, Parsing with compositional vector grammars, in: In Proceedings of the ACL conference, 2013, p. 455465.

[30] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, C. Potts, Recursive deep models for semantic compositionality over a sentiment treebank, in: Proceedings of the conference on empirical methods in natural language processing (EMNLP), Vol. 1631, Citeseer, 2013, p. 1642.

[31] P. S. Madhyastha, X. Carreras, A. Quattoni, Prepositional phrase attachment over word embedding products, in: Proceedings of the 15th International Conference on Parsing Technologies, 2017, pp. 32–43.

[32] P. Lauren, G. Qu, F. Zhang, A. Lendasse, Clinical narrative classification using discriminant word embeddings with elm, in: Neural Networks (IJCNN), 2016 International Joint Conference on, IEEE, 2016, pp. 2931–2938.

[33] A. Conneau, H. Schwenk, L. Barrault, Y. Lecun, Very deep convolutional networks for text classification, in: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, Vol. 1, 2017, pp. 1107–1116.

[34] P. Liu, X. Qiu, X. Huang, Recurrent neural network for text classification with multi-task learning, in: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16), 2016, pp. 2873–2879.

[35] S. Baker, A. Korhonen, S. Pyysalo, Cancer hallmark text classification using convolutional neural networks, in: Proceedings of the Fifth Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM 2016), 2017, pp. 1–9.

[36] N. V. Chawla, N. Japkowicz, A. Kotcz, Special issue on learning from

imbalanced data sets, ACM Sigkdd Explorations Newsletter 6 (1) (2004) 1–6.

[37] G. M. Weiss, Mining with rarity: a unifying framework, ACM Sigkdd Explorations Newsletter 6 (1) (2004) 7–19.

[38] M. Wasikowski, X. w. Chen, Combating the small sample class imbalance problem using feature selection, IEEE Transactions on knowledge and data engineering 22 (10) (2010) 1388–1400.

[39] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, F. Herrera, A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 42 (4) (2012) 463–484.

[40] N. Chawla, N. Japkowicz, A. Kotcz, special issue on learning from imbalanced data sets, ACM SIGKDD Explorations Newsletter 6 (1) (2004) 1–6.

[41] Z. Zheng, X. Wu, R. Srihari, Feature selection for text categorization on imbalanced data, ACM Sigkdd Explorations Newsletter 6 (1) (2004) 80–89.

[42] C. Ling, V. Sheng, Cost-sensitive learning and the class imbalance problem, Encyclopedia of Machine Learning.

[43] P. Juszczak, R. P. Duin, Uncertainty sampling methods for one-class classifiers, in: Proceedings of the Workshop on Learning from Imbalanced Datasets II, ICML, Vol. 3, 2003, p. 8.

[44] M. Moro, M. Tejedor, J. E. Hernandez, M. M. Garcia, A. R. Ingelmo, C. V. Albelda, Incidence of anaphylaxis and subtypes of anaphylaxis in a general hospital emergency department, J Investig Allergol Clin Immunol 21 (2) (2011) 142–149.

[45] M. Tejedor, M. Moro, M. García, J. Esteban Hernandez, A. Rosado Ingelmo, C. Vila Albelda, C. Gomez Traseira, R. Cardenas Contreras,

J. Sanz Sacristan, A. Hernandez Merino, Incidence of anaphylaxis in the city of alcorcon (spain): a population-based study, Clinical & Experimental Allergy 42 (4) (2012) 578–589.

[46] D. G. Altman, Practical statistics for medical research, CRC press, 1990.

[47] G. Salton, A. Wong, C.-S. Yang, A vector space model for automatic indexing, Communications of the ACM 18 (11) (1975) 613–620.

[48] C. Cardellino, Spanish Billion Words Corpus and Embeddings (March 2016).
URL http://crscardellino.me/SBWCE/

[49] J. Tiedemann, L. Nygaard, The opus corpus-parallel and free: http://logos.uio. no/opus., in: Proceedings of the Second Language Resources and Evaluation Conference (LREC 2004), 2004, pp. 1183,1187.

[50] M. Taulé, M. A. Martí, M. Recasens, Ancora: Multilevel annotated corpora for catalan and spanish., in: Proceedings of the Sixth Language Resources and Evaluation Conference (LREC 2008), 2008, p. 96101.

[51] M. F. Porter, An algorithm for suffix stripping, Program 14 (3) (1980) 130–137.

[52] O. Levy, Y. Goldberg, Dependency-based word embeddings., in: ACL (2), 2014, pp. 302–308.

[53] S. Wang, J. Jiang, Learning natural language inference with lstm, in: Proceedings of NAACL-HLT, 2016, pp. 1442–1451.

[54] J. MacQueen, et al., Some methods for classification and analysis of multivariate observations, in: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, Vol. 1, Oakland, CA, USA, 1967, pp. 281–297.

[55] P. Langley, W. Iba, K. Thompson, et al., An analysis of bayesian classifiers, in: Aaai, Vol. 90, 1992, pp. 223–228.

[56] V. Vapnik, The nature of statistical learning theory, Springer science & business media, 1995.

[57] S. H. Walker, D. B. Duncan, Estimation of the probability of an event as a function of several independent variables, Biometrika 54 (1-2) (1967) 167–179.

[58] N. S. Altman, An introduction to kernel and nearest-neighbor nonparametric regression, The American Statistician 46 (3) (1992) 175–185.

[59] F. Rosenblatt, Principles of neurodynamics. perceptrons and the theory of brain mechanisms, Spartan Book, 1962.

[60] L. Breiman, Random forests, Machine learning 45 (1) (2001) 5–32.

[61] Y. LeCun, P. Haffner, L. Bottou, Y. Bengio, Object recognition with gradient-based learning, in: Shape, contour and grouping in computer vision, Springer, 1999, pp. 319–345.

[62] T. Joachims, Text categorization with support vector machines: Learning with many relevant features, Machine learning: ECML-98 (1998) 137–142.

[63] J. Kivinen, M. K. Warmuth, The perceptron algorithm vs. winnow: linear vs. logarithmic mistake bounds when few input variables are relevant, in: Proceedings of the eighth annual conference on Computational learning theory, ACM, 1995, pp. 289–296.

[64] V. Nair, G. E. Hinton, Rectified linear units improve restricted boltzmann machines, in: Proceedings of the 27th international conference on machine learning (ICML-10), 2010, pp. 807–814.

[65] T. M. Oshiro, P. S. Perez, J. A. Baranauskas, How many trees in a random forest?, in: MLDM, Springer, 2012, pp. 154–168.

[66] D. Zeng, K. Liu, S. Lai, G. Zhou, J. Zhao, Relation classification via convolutional deep neural network, in: Proceedings of COLING 2014, the 25th

International Conference on Computational Linguistics: Technical Papers, 2014, pp. 2335–2344.

[67] S. Lai, L. Xu, K. Liu, J. Zhao, Recurrent convolutional neural networks for text classification., in: Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15), Vol. 333, 2015, pp. 2267–2273.

[68] J. P. Chiu, E. Nichols, Named entity recognition with bidirectional lstm-cnns, Transactions of the Association for Computational Linguistics 4 (2016) 357370.

[69] M. Artetxe, G. Labaka, E. Agirre, K. Cho, Unsupervised neural machine translation, in: Proceedings of the Sixth International Conference on Learning Representations, 2018.

[70] W. Che, Y. Zhang, Deep learning in lexical analysis and parsing, in: Deep Learning in Natural Language Processing, Springer, 2018, pp. 79–116.

[71] J. L. Elman, Finding structure in time, Cognitive science 14 (2) (1990) 179–211.

[72] S. Lai, L. Xu, K. Liu, J. Zhao, Convolutional neural networks for sentence classification, in: 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, p. 17461751.

[73] J. Wang, Z. Wang, D. Zhang, J. Yan, Combining knowledge with deep convolutional neural networks for short text classification, in: Proceedings of the 26th International Joint Conference on Artificial Intelligence, AAAI Press, 2017, pp. 2915–2921.

[74] W. B. Frakes, R. Baeza-Yates, Information retrieval: data structures and algorithms, Prentice Hall PTR, 1992.

[75] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: International Conference on Learning Representations (ICLR), 2015, pp. 1–13.