Machine Learning for Drug Efficiency Prediction

Hafida Tiaiba[1(✉)], Lyazid Sabri[1,2], Abdelghani Chibani[2], and Okba Kazar[3,4]

[1] Maths and Informatics Faculty, Mohamed El Bachir El Ibrahimi University, Bordj Bou Arreridj, Algeria
{hafida.tiaiba,lyazid.sabri}@univ-bba.dz

[2] LISSI-The Laboratory of Images, Signals and Intelligent Systems, University Paris-Est Vitry-sur-Seine, Île-de-France, France
{sabri,chibani}@lissi.fr

[3] LINFI Laboratory, Computer Sciences Department, University of Biskra, Biskra, Algeria
o.kazar@univ-biskra.dz

[4] Department of Information Systems and Security, College of Information Technology, United Arab Emirates University, Al Ain, United Arab Emirates
o.kazar@uaeu.ac.ae

Abstract. Health-related social media data, particularly patients' opinions about drugs, have recently provided knowledge for research on the adverse reactions, allergies that a patient experiences and drug efficacy and safety. We develop an effective method for analyzing medicines' efficiency and conditions-specific prescription from patient reviews provided by Drug Review Dataset (drug.com). Our approach relies on the Natural Language Processing (NLP) principle and a word embedding vectorization method to preserve semantics. For this purpose, we conducted experiments using various sampling techniques, precisely random sampling and balanced random sampling. Furthermore, we applied several statistical models: Logistic Regression, Decision Tree, Random Forests, K-Nearest Neighbors (KNN) and Neural Network models (simple perceptron, multilayer perceptron and convolutional neural network). We varied the size of training and test data sets to study the effect of the sampling techniques on model efficiency. Compared to other models, the results show that the proposed models in this paper: KNN, Embedding-100, and CNN-Maxpooling outclass models proposed by several researchers. Indeed, Embedding-100 has achieved better training accuracy and test accuracy. Moreover, during our study, we concluded that different factors influence the effectiveness of the models, mainly the text preprocessing method, sampling techniques in terms of size and type, text vectorization method and machine learning models.

Keywords: Machine Learning · Text Classification · Word Embedding · Health · Predict Drug Efficiency · Natural Language Processing

A. Cunha et al. (Eds.): MobiHealth 2022, LNICST 484, pp. 322–337, 2023.
https://doi.org/10.1007/978-3-031-32029-3_27

1 Introduction

The prescription of a drug is crucial; an initial step in the circuit in the personalization of diagnosis and treatment helps therapeutic education and promotes more effective precision medicine. The effectiveness of a cure is closely linked to the positive results obtained by minimizing the adverse effects. Nevertheless, the efficacy of a drug relies on an important criterion, namely, whether a patient strictly adheres to the dosing schedule dosage. Treatment may work well in clinical trials but may not produce the same results in the real world due to side effects. That is why patients stop taking this drug without a doctor's advice. Implementing new technologies for patient well-being depends on many factors, such as personal experience, patient practices and expert opinion. One can also ask whether patients' reviews (i.e., feedback) on medicine can help evaluate drugs. The High Authority of Health (HAS) [1] answers this question for Health. HAS assesses health technologies from a clinical and medico-economic point of view. HAS has been integrating patients into the drug evaluation process for several years (November 2016). The result is clear, the HAS advocates collaboration between health actors and patients.

Among the HAS surveys, a pharmaceutical association observed that patients resorted to cutting a pill at the level of a groove for lack of drugs and inadequate doses. However, it is not possible to regularly break a tablet; therefore, getting the required dose becomes difficult. Thus, everyone's participation (e.g., patients and pharmacists) in audits and/or assessments made available to doctors is an asset for the patient's well-being. Analyzing side effects and the effectiveness of drugs based on opinions and feedback from patients is our approach's objective. The results will aid the medical staff in better preventing and providing care. The proposed approach relies on the following mechanisms and techniques:

1. Investigate the impact of dataset sampling on model performance.
2. Analyze the effect of preprocessing method on model accuracy.
3. Apply the word embedding principle for reasoning with the textual semantic in formation.
4. The use of data mining models, statistical models, and neural network models to compare the two types of models and select the best model.

Natural Language Processing (NLP) is a field of Computer Science and Artificial Intelligence. NLP is either a rule-based or statistical-based technique. In addition, NLP and machine learning can analyze large volumes of data written in natural language [2]. We proceeded to textual classification to extract knowledge from patients' narrative opinions on drugs for a specific disease. This approach aims to find a better textual classification model based on NLP. It automatically assigns an opinion on whether or not the drug used for a specific disease is effective. The results should provide healthcare partners, pharmacists, physicians, and managers with valuable drug information.

A class assignment relies on the drug's designation, the type of disease, and the patient's opinion. We have opted for two strategies to prepare the data: The first preserves the semantics of the texts (i.e., does not modify the content of the reviews written

in natural language). The second strategy relies on the lemmatization principle. Moreover, instead of representing each word using a numeric vector using the one-hot representation, we have described each document in unstructured (narrative) text format using a numeric vector. To increase the performance of our approach, we used the word embedding representation, in which similar words have an identical encoding, using the Embedding layer of Keras [2]. Finally, we used data mining algorithms for data analysis.

Previous works on analyzing patient reviews of medications have focused on improving learning methods. However, the document preprocessing phase (i.e., text cleaning) differs slightly from traditional methods. This phase influences the classification result and the medical textual data since the sentences-preprocessing can lead to the deletion of words and contexts and bias the meaning. In addition, a medical (clinical) record is a sequence of structured chronological elementary events (i.e., the dynamic narrative of the patient's history). Thus, lexical processing via lemmatization can alter temporal knowledge. Indeed, reporting all the past tenses to their corresponding infinitive tenses will change the meaning of a patient's medical history. The work carried out by Bemila et al. [3] and Mascio et al. [4] relied on removing stop words, special characters and punctuation, and then the authors applied lemmatization and stemming. The only difference between the two authors is that Mascio, Aurelie, et al. used SciSpacy during the lemmatization process. Unlike Bemila, Mercadier, Yves [5] did not use stemming. Gräber et al. [6], Vijayaraghavan, and Basu [7] preferred to delete the terms whose frequency is lower than a threshold in addition to the preprocessing phase. In [6], the authors used Drug.com and applied the logistic regression classifier.

In this work, we use Drug.com data sets that include 215,063 patient reviews. We compare different statistical models, namely LogisticRegression, RandonForest, DecisionTree, KNeighbors and neural network models: Embedding-100, Simple10, Embedding-32-, Embedding-32-Maxpooling and CNN-Maxpooling. We also studied the effect of sampling training and test datasets and the text-cleaning process on the effectiveness of the proposed models. Furthermore, we analyze the impact of semantic reasoning using word embedding on the performance of neural network models. Finally, we also analyze the effect of the dimension size of the vectors resulting from word embedding on the efficiency of the models. The automatic analysis of the evaluations of patient opinions made by our study makes it possible to obtain relevant information on side effects, drug interactions, and the effectiveness of drugs.

To our knowledge, the proposed model allows for determining drug efficiency using drug.com data sets is the first to achieve 0.9984 accuracies. For example, the following opinion is categorized as negative (i.e., this class indicates that the drug used has side effects): "This medicine does not work for me, each time I take it, it makes me drowsy, but I could not sleep". The following patient's opinion, "It has no side effects, I take it in combination of Bystolic 5 Mg and Fish Oil", has been labelled positive (i.e., positive opinion) and therefore allows inferring that the drug is efficient.

This paper is organized as follows: Sect. 2 presents state-of-the-art. Section 3 describes our approach to classifying large-scale drug.com data. We also detail the pseudo algorithms we developed in this section with the results. Finally, Sect. 4 concludes the paper and proposes new research directions and perspectives.

2 Related Work

Several studies have been devoted to information representing the experiences and opinions of consumers. For example, the authors Jiménez-Zafra et al. have examined how patients express their opinions in the medical forum [8]. They aim to determine the best way to exploit sentiment analysis in this area. They applied supervised learning and lexical sentiment analysis approaches to two different corpora extracted from the social web. Among the datasets most used in studies on the opinion of patients on drugs is drug.com. Most of the studies were based only on the patients' opinions for the classification of their reviews.

However, those studies ignored the drug designation and conditions. Patient reviews are characterized by a combination of informal language with specific terminology and a high degree of lexical diversity, making it more challenging to analyze. The authors in [9] proposed a text vectorization method based on Term Frequency-Inverse Document Frequency (TF-IDF) and FastText word embeddings. They used five classifiers: Support Vector Machine (SVM), Decision Tree, Random Forests, Naïve Bayes, and K-nearest neighbors. Colón-Ruiz et al. proposed a system based on word embedding, and Long short-term memory (LSTM) classifier. Their approach achieved an accuracy of 0.9046 compared to other models they studied, such as Convolutional Neural Network (CNN) and CNN combined with LSTM [10]. As for Gräber et al. studied the performance of their models on the data on a single condition [6]. Then they evaluated the models on other subsets related to the conditions. Specific conditions are selected by extracting five of the most common disorders in the Druglib.com dataset. Such as Depression, Pain, Anxiety, Contraception and Diabetes Type 2. The logistic regression model applied to drug.com achieved an accuracy of 0.9224. In [11], the authors used the principle of TF-IDF and obtained an accuracy of 0.316 achieved by the Roberta classifier with the Data Augmentation and Inference Augmentation (DAIA) method. While in [3], the authors applied the Bag of Words principle and used Naive Bayes, Random Forest, Linear Support Vector Classification (SVC), Logistic Regression and RNN-BiLSTM (Recurrent neural networks - A Bidirectional Long Short-Term Memory) algorithms. MIN, Z. used AskaPatient data sets and proposed a combined WSM-CNN-LSTM (Weakly Supervised Mechanism) model of CNN and LSTM. His model provided an accuracy of 86.72 [12].

3 Classification Approach

3.1 Theoretical Study

We recall here our approach for a better prediction of drug intake. Thus, we considered different criteria, such as the size of the samples, the semantic contribution and the study of the deep-learning and statistical models. The process of our research is as follows:

1. Relying on the binary classification to predict patient satisfaction with medication use and perceptions of side effects and efficacy.
2. Study the impact of data set sampling on model performance.
3. Analyze the effect of preprocessing method of model accuracy.
4. Apply word embedding for the semantic reasoning of texts.
5. Compare statistical models and artificial neural network models.

A Machine Learning Dataset. We chose the drug.com dataset from the UCI Machine Learning Repository [13]. It includes 215,063 samples presented in two files (training and test).

Each sample has the following fields: drug name and a drug's number, terms that describe the reason for using the drug or the patient's illness, user reviews, ratings given by the user, the date on which the drug was reviewed, and the number of users who found the review useful.

In this paper, we concatenated the two files (drug_train, drug_test), and then we reconstituted the training and test data set by randomly choosing the documents for two case studies.

For the first case, we selected 75% for training and 25% for the test. While for the second case, we used 80% for training and 20% for the test. We also carried out random and random balanced sampling. The second sampling method aims to maintain the proportion of ranking values across the training and testing sets. Our approach relies on the 'stratify' parameter of the 'train_test_split' function [14]. To evaluate the certainty of patient satisfaction, we derived two level labels for patient satisfaction. After that, we attributed positive classification "1" for a rating interval from 5 to 10 points and negative classification "0" for a rating less than five to one.

Data Preprocessing Strategies. Two strategies have been conducted; the first consists in carrying out the transformations in the sentences written in natural languages. All uppercase characters are transformed to lowercase, removing the unique character ("). Then, the spaces are deleted in the name of the drug and the conditions. The goal is to treat each as a single word (i.e., each of these data will be represented by a number). Therefore, after concatenating the drug name, condition and patient's feedback, their one-hot representations will be straightforward as long as a single number describes each word. We highlight that during the process of this first strategy, we chose to discard the stemming and lemmatization process, avoid removing punctuation marks and numbers from texts.

Removing the question's marks, dots, commas, and numbers such as '5 Mg' and '10 Mg' will change the meaning of a patient's medical history. As for a stemming process that reduces or removes a part of a word, which can completely distort the sentence's meaning, for example, the implications of the words 'suffering', 'suffered' and 'suffer' are different since the stemming result gives the same stem; hence the meaning of the sentence will be biased. At the end of this strategy's process, the obtained vocabulary size is 84,362 words, and the document's length is 2,405. While in the second strategy, we opted for the principle of lemmatization based on the Natural Language Toolkit (NLTK) library associated with the Wordnet dictionary. In this case, the vocabulary's length is 83,162 and as in the first strategy, we obtained the exact size of the longer document.

Converting text to numerical data allows describing each document by a vector instead of representing each word by a numerical vector. Since each document includes the name of the drug, the conditions, and the patient comments, and considering the numeric vectors of different sizes, we applied the padding principle to make all vectors the same length. The size of the document vectors is 2,405. Concerning the statistical classifiers, we used only one-hot; for the neural classifiers, we used One-hot and word embedding of different dimensions (32 and 100) to analyze the effect of vectorization on model performance.

A Proposed Models and Algorithms for Drug Efficiency Prediction. We opted for logistic regression, random forest, decision trees, and K-nearest neighbors among the statistical models, for this purpose we used sklearn library [15]. Concerning the random forests, we set 'n_estimators', the parameter allowing us to define the number of trees in the random forest, to 10 and the maximum depth of each tree 'max_depth' to 3. Same thing for the decision tree; we limit the maximum depth 'max_depth' to 3. For the K-nearest neighbors' models, we used the Euclidean distance and set the number of neighbors to 1.

For the neuronal models, we used the sequential model of the Keras library [16]. The latter provides a way to convert positive integer representations of words to word embedding by an embedding layer. In word embedding, words are encoded by numeric value vectors in high-dimensional space, where the similarity between words translates to proximity in vector space for semantic reasoning.

Each document that includes a drug name, condition, and patient comment is represented by One–hot. Then, relying on the embedding layer, a vector of the real numbers represents each document. Keras' embedding layer allows choosing the dimension of the vector representing the document. We treated the case of an Embedding with a size of 32 for the vectors representing the model (Embedding-32-). Also, for the Embedding-100 model, we used a representation by word embedding of dimension 100. We have proposed the artificial neural models: Embedding-100, Embedding-32-, Embedding-32-Maxpooling and CNN-Maxpooling. Embedding-100 is a simple perceptron consisting of a single neuron with an embedding layer of dimension 100. A vector of size 100 represents each word, allowing the representation of the words efficiently and densely in which similar words have similar encoding [17]. Therefore, each document will be represented by a vector of two size dimensions (2,405 and 100).

This process allowed us to apply the flatten principle to create a single vector of one dimension (240 500) to be directly used with the next fully connected layer. Simple10: is a neural network with a hidden layer consisting of 10 neurons linked to a single output neuron using One-hot representation. The Embedding-32- model has the same structure as Simple10 but with an additional Embedding layer. The latter has a dimension of 32.

We applied the Global max-pooling on the Embedding-32 model for building the Embedding-32-max-pooling model. Finally, the proposed CNN-Maxpooling model: is a convolutional neural network (CNN) composed of a one-dimensional convolution layer (Conv1D). The number of filters is 128; the kernel size is 5. We applied an embedding of size 32 and then a global max-pooling (GlobalMaxPool1D). GlobalMaxPool1D takes the maximum of each vector, which reduces the dimension of the resulting embedding matrix.

For all artificial neural models, the activation function used is Rectified Linear Unit Activation Function (ReLU) and 'sigmoid' for the output layer. Concerning the Embedding-100, we applied the sigmoid function exclusively because we have a binary classification. Furthermore, we rely on the optimization 'adam' implementation of the gradient descent function [18], and the 'binary_crossentropy' loss function suited to binary classification problems. In addition, the parameters 'metrics' is 'accuracy', representing the accuracy of the models used to evaluate the different models. Accuracy measures the labels assigned by our approach. Formally, accuracy is handled by the

following formula:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

where, we use TP (True Positive), TN (True Negative), FP (False Positive) and FN (False Negative).

The pseudo-algorithm, 'General Pseudo Algorithm', synthesizes our approach and handles the preprocessing, sampling, vectorization, and model training, Table 1.

According to the obtained results, we chose the Embedding-100 model. The parameters of this model are:

– The sigmoid (X) activation function is: $\frac{1}{1+e^{-x}}$

Table 1. General Pseudo Algorithm.

```
Data1= Drug_train ∪ drug_test{union}Conversion "review" to lowercase, removing (")
Removal of drug name spaces
Removal of condition spaces
Data_drug['review'] =  concatenation of (drug name, condition and review)
if (Sentiment >= 5) then  Review_Sentiment=1
else Review_Sentiment=0
endif
/* Data_drug: contain review and  Review_Sentiment */
Split Drug_train into sample_set = {s_75, s_75_stratify, s_75_lemmatization,
s_75_lemmatization_stratify, s_80, s_80_stratify, s_80_lemmatization,
s_80_lemmatization _stratify}
if sample in (s_75_lemmatization, s_80_lemmatization,
    s_75_lemmatization_stratify, s_80_ lemmatization_stratify) )
 then
     Lemmatization of review
endif
D=Data_drug['review']  /* the corpus*/
Di=(di1,di2, … ,din)   /* ( n<=2405 longest document,  i<=215063 )  */
Y= Data_drug['Review_Sentiment'] { labels}
R= One_hot(D) / Ri= (Ri1,Ri2,      ,Rin)   : numeric representation
P= padding(R) / Pi=(Pi1,Pi2,     ,Pi2405): make all vectors of the same size (2405) .....(a)
Set_model={   LogisticRegression,   DecisionTree,   KNeighbors,   Embedding-100,
Simple10, Embedding-32-, Embedding-32-Maxpooling, CNN-Maxpooling }
For every  sample in sample_set do
     For  every model in Set_model do
              Train model
              Test model
     Endfor
Endfor
Best model = model where max (train accuracy, test accuracy)
End
```

Table 2. Pseudo Algorithm for generating Embeddig-100 (the best model).

Input : P ∈ R^{240500} Calculated in General Algorithm step (a) sample = s_75_stratify (75% train 25% test. case balanced random of data_drug without lemmatization) , Y= review_Sentiment
Output:Trained model
Begin
{Train the model: 75% of dataset}
Initialize w_0, $W_i \in R^{240500}$
For each training example $(P_i,yi) \in (P,Y)$
E_i= embedding(P_i)/ E_i is matrix(2405,100)
F_i= flatten(E_i) / F_i =(E_{i1}, E_{i2}, , E_{i3}, , $E_{i240500}$.)/ One dimension
yi'= sigmoid($W_{it}^T \times F_i + w_0$) ……..…(b)
/* W_{it}^Ttransposed vector of W_{it}*/
if yi ≠ yi' then calculate loss (yi , yi') given in (3)
update W_{it} using dam optimizer given in formula (4)
go to (b)
endif
Endfor
{Test the model : 25% of dataset }
Initialize w_0, $W_i \in R^{240500}$
For each training example $(P_i,yi) \in (P,Y)$
E_i= embedding(P_i)/ E_i is matrix(2405,100)
F_i= flatten(E_i) / F_i =(E_{i1}, E_{i2}, , E_{i3}, , $E_{i240500}$.)/ One dimension
yi'= sigmoid($W_{it}^T \times F_i + w_0$) ……..…(c)
/* W_{it}^Ttransposed vector of W_{it}*/
if yi ≠ yi' then calculate loss (yi , yi') given in (3)
update W_{it} using dam optimizer given in formula (4)
go to (c)
endif
Endfor
Return final weights vector
Save the model: Embedding_100_75final.h5
(to use it in Predictions)
End

– The performance of a classification model, the binary cross-entropy loss function is handled by the following formula:

$$L = -\frac{1}{output\ size} * X. \tag{2}$$

$$X = \sum_{i=1}^{out\ put\ size} y_i . log\, y'_i + (1 - y_i) . \log(1 - y'_i)$$

$$L = -y.\log y' + (1 - y).\log(1 - y') \tag{3}$$

where: y' is the label computed by the model, and y is the predicted label. Adam optimiser is used as follows for updating weights [19]:

$$w_{i,t+1} = (w_{i,t} - \alpha.m_t) \tag{4}$$

where: $m_t = \beta m_{t-1} + (1 - \beta)\frac{\partial L}{\partial W_{i,t}}$

m_t: aggregate of gradients at iteration t initially $m_t = 0$,
α: learning rate,
$W_{i,t}$: weights at time t,
$W_{i,t+1}$: weights at iteration $t + 1$,
∂L: derivation of loss function,
$\partial W_{i,t}$: derivation of weights at iteration t,
β: moving average parameter,

The pseudo algorithm illustrated in Table 2 explains the sequence of steps of the Embedding_100 model approach.

To exploit our Embedding_100 model, we use the Prediction pseudo algorithm presented in Table 3 that makes predictions regarding drug reviews.

Table 3. Prediction Pseudo Algorithm.

```
Input  :  Drug name, Condition, Patient review
Output: drug efficiency
Begin
  Enter drug name and condition
  Enter review
  Conversion patient review to lowercase
  Removal of drug name spaces
  Removal of condition spaces
  review = concatenation of (drug name, condition and review)
  r=one_hot(review)
  p=padding(r)
  Embedding_100_75final(p)
  If  y=0 then  'the drug presents conflit'
      else ' the drug presents no conflit'
End
```

3.2 Practical Study

Example Illustrating Stages in Natural Language Processing. Consider the following sentences taken from the drug.com: The first approach transforms the text into lowercase and removes spaces in the name of the drugs and the instructions for use (i.e.,

Table 4. Text Vectorization.

Approach 1: The length of the largest document is 2,405 and the vocabulary size is 84,362
Azithromycin ChlamydiaInfection was prescribed one dose over the course of one day, took 4 pills of 250mg after a light lunch, and had nausea and mild stomach pains/upset. lying down did not alleviate the discomfort and threw up 3 hours later. called up my doctor to check if i needed to take another dose but he said my body would have absorbed the pills by then. still experiencing mild stomach pains but nausea is mostly gone now.stomach pains but nausea is mostly gone now.
One hot : [13084, 50875, 62892, 38293, 1436, 6294, 46079, 26091, 52962, 4736, 1436, 38726, 65804, 33200, 18587, 4736, 78333, 10937, 8946, 10470, 70093, 36469, 31023, 20051, 36469, 43536, 9610, 83427, 82756, 7416, 69605, 45744, 66264, 34414, 26091, 6303, 36469, 52222, 13236, 78384, 76910, 58260, 49736, 13236, 61167, 74271, 77306, 65408, 43571, 65228, 25474, 77306, 55872, 16926, 6294, 65610, 53047, 1232, 61167, 23363, 48658, 15209, 57011, 26091, 18587, 10796, 70690, 22086, 32677, 43536, 9610, 83427, 65610, 20051, 68236, 9373, 49136, 75495]
Padding : [13084 50875 62892 ... 0 0 0]
Embedding : tf.Tensor([[-0.00030909 0.04441366 -0.01012845 ... -0.01254865 -0.02349045 -0.03326954] [-0.03938956 0.0266563 -0.02233372 ... 0.02533278 0.04644166 0.01073159]... [-0.00268913 -0.03778163 -0.04249182 ... 0.04519937 -0.00611898 0.04797543] [-0.00268913 -0.03778163 -0.04249182 ... 0.04519937 -0.00611898 0.04797543]], shape=(2405, 100), dtype=float32)
Flatten: [-0.00030909 0.04441366 -0.01012845 ... -0.01254865 -0.02349045 -0.03326954 -0.03938956 0.0266563 -0.02233372 ... 0.02533278 0.04644166 ... 0.04519937 -0.00611898 0.04797543 -0.00268913 ... 0.04519937 -0.00611898 0.04797543]
Approach 2: The length of the largest document is 2,405 and the vocabulary size is 83,162
Azithromycin ChlamydiaInfection wa prescribed one dose over the course of one day, took 4 pill of 250mg after a light lunch, and had nausea and mild stomach pains/upset. lying down did not alleviate the discomfort and threw up 3 hour later. called up my doctor to check if i needed to take another dose but he said my body would have absorbed the pill by then. still experiencing mild stomach pain but nausea is mostly gone now.
One hot : [12172, 39241, 39536, 68005, 22878, 48912, 81513, 17567, 70694, 44772, 22878, 17104, 56100, 42384, 25484, 44772, 5599, 49865, 2936, 81692, 4995, 64433, 15463, 26371, 64433, 22062, 57770, 26197, 31686, 20524, 61211, 77066, 29682, 70526, 17567, 23297, 64433, 8632, 57916, 6464, 71041, 11634, 8126, 57916, 46389, 58551, 35636, 79304, 52635, 13974, 9880, 35636, 48878, 22702, 48912, 25530, 31781, 81052, 46389, 72903, 79338, 80701, 56709, 17567, 25484, 24160, 75320, 72300, 63521, 22062, 57770, 51437, 25530, 26371, 52854, 16037, 48754, 16579]
Padding : [12172 39241 39536 ... 0 0 0] Embedding: tf.Tensor([[0.03626031 -0.00316075 0.0112661 ... -0.02691853 -0.03584041 0.00848019] [-0.01687868 -0.03148886 -0.03002917 ... -0.03701594 -0.03470857 -0.01378261]... [0.02724567 0.01433365 -0.01567432 ... -0.00919002 -0.03320863 -0.00234286] [0.02724567 0.01433365 -0.01567432 ... -0.00919002 -0.03320863 -0.00234286]], shape=(2405, 100), dtype=float32)
Flatten: [0.03626031 -0.00316075 0.0112661 ... -0.02691853 -0.03584041 0.00848019 -0.01687868 -0.03148886 -0.03002917 ... -0.03701594 -0.03470857 -0.01378261 0.01509024

(*continued*)

Table 4. (*continued*)

-0.04584317 -0.031525 ... -0.00919002 -0.03320863 -0.00234286]
Approach 3: The length of the largest document is 1,036 and the vocabulary size is 61,318
azithromycin chlamydiainfect wa prescrib one dose cours one day took pill mg light lunch nausea mild stomach painsupset lie not allevi discomfort threw hour later call doctor check need take anoth dose said bodi would absorb pill still experienc mild stomach pain nausea mostli gone
One hot : [45626, 28111, 3375, 52534, 31538, 28868, 10091, 31538, 57765, 34662, 32504, 18006, 58030, 12401, 21328, 22967, 23849, 15936, 40673, 49312, 40475, 30382, 49223, 54143, 4428, 4676, 14683, 7214, 6879, 2871, 10144, 28868, 15097, 15031, 52534, 12338, 32504, 26673, 39566, 22967, 23849, 51801, 21328, 26871, 13077]
Padding : [45626 28111 3375 ... 0 0 0]
Embedding: tf.Tensor([[-0.03886646 -0.0198079 0.01174947 ... -0.03995751 -0.04186364 0.01677601] [-0.04668018 -0.04458895 -0.00548612 ... -0.01382007 0.01983864 -0.03390007]... [0.03535514 -0.04283841 -0.04564927 ... 0.031701 -0.03638158 0.0310008] [0.03535514 -0.04283841 -0.04564927 ... 0.031701 -0.03638158 0.0310008]], shape=(1036, 100), dtype=float32)
Flatten: [-0.03886646 -0.0198079 0.01174947 ... -0.03995751 -0.04186364 0.01677601 -0.04668018 -0.04458895 -0.00548612 ... –0.01382007 0.01983864 -0.03390007 0.04955867 0.03802111 0.03902782 0.03535514... 0.031701 -0.03638158 0.0310008]

conditions). As for the second approach, we likewise integrate the process of lemmatization. As the majority of the state-of-the-art learning methods, we remove symbols, special characters, numbers, and punctuation in the third approach. Furthermore, we use lemmatization and stemming. The experiments highlight those two approaches' limitations that cause the general loss of vital knowledge in health care. For clarity, the words (i.e., term) emphasized with yellow are deleted in the third or second approach. Otherwise, the terms highlighted in grey color are edited (i.e., modified) within the third and second approaches. We have exploited only the first and second approaches in this study.

An example of the different types of text vectorization in different approaches is presented in Table 4 to show the impact of pre-processing on the text, in particular, the semantics of the text.

Experimental Results. We studied the models' performance according to three criteria: The classifier's impact, the sampling method, and the size of the training and test data sets.

Discussion. The first strategy shows that in the case of a random sampling of 75% for training and 25% for testing, the Embedding-100 classifier achieved better results with a training accuracy of 99.83%, Table 5. This result is similar to the Embedding-32- (99.72%), and CNN-Max-pooling (99.71%) models. Within the random sampling process, the train's accuracy increased for all models except for the decision tree model. In the case of a random sampling of 80% for training and 20% for testing, the train's accuracy value increases by 0.0026 for the KNeighbors model. The best test accuracy

Table 5. Models' accuracy in case of random sampling without lemmatization.

Model	75% train 25% test		80% train 20% test	
	Train accuracy	Test accuracy	Train accuracy	Test accuracy
LogisticRegression	0.7508	0.7512	0.7506	0.7521
RandonForest	0.7508	0.7512	0.7506	0.7521
DecisionTree	0.7525	0.7534	0.7506	0.7521
KNeighbors	0.9960	0.7928	0.9986	0.8037
Embedding-100	0.9983	0.9983	0.9983	0.9983
Simple10	0.7508	0.7512	0.7506	0.7521
Embedding-32-	0.9972	0.9011	0.9976	0.9083
Embedding-32-Maxpooling	0.9176	0.8591	0.9236	0.8703
CNN-Maxpooling	0.9971	0.9322	0.9973	0.9365

Table 6. Models' accuracy in case of balanced random sampling ('stratify' parameter) without lemmatization.

Model	75% train 25% test		80% train 20% test	
	Train accuracy	Test accuracy	Train accuracy	Test accuracy
LogisticRegression	0.7509	0.7509	0.7509	0.7509
RandonForest	0.7509	0.7509	0.7509	0.7509
DecisionTree	0.7509	0.7509	0.7509	0.7509
KNeighbors	0.9986	0.7938	0.9986	0.8032
Embedding-100	0.9984	0.9984	0.9984	0.9984
Simple10	0.7509	0.7509	0.7509	0.7509
Embedding-32-	0.9961	0.8937	0.7509	0.7509
Embedding-32-Maxpooling	0.9152	0.8638	0.9179	0.8664
CNN-Maxpooling	0.9972	0.9340	0.9970	0.9347

(0.9984) is achieved by the Embedding_100 model for the two sampling cases (75% and 80% for training and 25% and 20% for test). Indeed, the CNN-Maxpooling model obtained 0.9972 (for 75%) and 0.9970 (for 80%) in the training case and test accuracy of 0.9340 and 0.9347. Therefore, the size of the training and test data sets impacts the models' performance. We notice well from the results that we have different results for most cases by changing the number of training and test data. Table 6 describes that the KNeighbors model (0.9986) obtains the best performance in the training case. The best test accuracy (0.9984) is achieved by the Embedding_100 model for the two sampling

Table 7. Models' accuracy in case of random sampling with lemmatization.

Model	75% train 25% test		80% train 20% test	
	Train accuracy	Test accuracy	Train accuracy	Test accuracy
LogisticRegression	0.7508	0.7512	0.7481	0.7491
RandonForest	0.7508	0.7512	0.7506	0.7521
DecisionTree	0.7537	0.7541	0.7506	0.7521
KNeighbors	0.9960	0.7782	0.9986	0.8054
Embedding-100	0.9982	0.9015	0.9984	0.9104
Simple10	0.7508	0.7512	0.7506	0.7521
Embedding-32-	0.9974	0.8996	0.7506	0.7521
Embedding-32-Maxpooling	0.9093	0.8610	0.9091	0.8626
CNN-Maxpooling	0.9960	0.9271	0.9962	0.9345

Table 8. Models' accuracy in case of balanced random sampling ('stratify' parameter) with lemmatization.

Model	75% train 25% test		80% train 20% test	
	Train accuracy	Test accuracy	Train accuracy	Test accuracy
LogisticRegression	0.7509	0.7509	0.7508	0.7508
RandonForest	0.7509	0.7509	0.7509	0.7509
DecisionTree	0.7509	0.7509	0.7523	0.7521
KNeighbors	0.9986	0.7745	0.9986	0.7924
Embedding-100	0.9984	0.9031	0.9983	0.9082
Simple10	0.7509	0.7509	0.7509	0.7509
Embedding-32-	0.9972	0.8966	0.9968	0.9026
Embedding-32-Maxpooling	0.9122	0.8632	0.9181	0.8683
CNN-Maxpooling	0.9963	0.9307	0.9965	0.9385

cases (75% and 80% for training and 25% and 20% for the test). Indeed, the CNN-Maxpooling model obtained 0.9972 (for 75%) and 0.9970 (for 80%) in the training case and test accuracy of 0.9340 and 0.9347.

Table 7 shows that in the case of data lemmatization and random sampling for 75% training data sets, the best accuracy obtained is 0.9982 for Embedding-100. While in the case of training data sets of 80%, we notice that the K-Neighbors model gets a slight performance evaluated to 0.9986 (i.e., a difference of 0.0004 compared to

Table 9. The best machine learning model.

Model	Train accuracy	Test accuracy	Sampling
Embedding-100	0.9983	0.9983	75% train 25% test random sampling without lemmatization
Embedding-100	0.9984	0.9984	80% train 20% test and 75% train 25% test balanced random sampling without lemmatization
KNeighbors	0.9986	0.8054	80% train 20% test random sampling with lemmatization

Embedding-100). In Table 8, the balanced random sampling case with lemmatization K-Neighbors achieved a better training accuracy of 0.9986. However, K-Neighbors model achieves very low test accuracy compared to Embedding-100, Embedding-32-, and CNN-Maxpooling. We concluded that, on the one hand, the use of GlobalMaxpooling negatively affects the accuracy of the Simple10.

On the other hand, lemmatization also causes a decrease in performance for the models: Embedding-32-Maxpooling and CNN-Maxpooling. With an exciting reduction for the Embedding-32- model in the case of 80% training data sets with random sampling, except for the K-Neighbors model (i.e., presents the stability of these results). As for word embedding, it has increased the performance of neural network models. It is important to note that the results show the effect of word embedding for the Simple10 model for all the study cases. We propose choosing the Embedding-100 model for the case, 75%–25%, with balanced random sampling without lemmatization. Indeed, this model has a training accuracy of 0.9984 and a test accuracy of 0.9984, Table 6.

In our comparative state-of-the-art learning models, however, we retained only the models having obtained better performances, as indicated in Table 9. Comparative studies on different deep learning architectures, such as recurrent short-term memory neural networks (LSTM) and convolutional neural networks (CNN), have been conducted by Colón-Ruiz and Segura-Bedmar [10]. This study highlighted the importance and performance of the combination of Bert and Word2vec representation-based models. All the techniques studied proceed by deleting numerical expressions.

Furthermore, Bemila et al. [3] tested several classifiers on bags of words, such as Naive Bayes, logistic regression and RNN-BiLSTM. The best accuracy score is 0.83906, obtained using an RNN-BiLSTM classifier. Additionally, Na and Wai [20] proposed a rule-based linguistic approach. Their method takes a purely linguistic approach to calculating the sentiment orientation of a clause from prior sentiment scores assigned to words, taking into account grammatical relationships and semantic annotation of words in the text. Their approach achieved an accuracy of 0.69.

Table 10 highlights that our approach overcame the logistic regression model proposed by F. Gräber et al. [6], whose precision was 0.9224. The proposed text preprocessing method relies on word embedding, which keeps the text's semantics. Based on our study's research, the models: KNeighbors, Embedding-100, Embedding-32- and CNN-Maxpooling proposed by our study achieved the best accuracy of all analysis and classification models for drug.com dataset.

Table 10. Comparison with previous studies.

Study	Year	Method	Accuracy
Mercadier, Yves [5]	2021	XLNet [a]	0.3678
S. Vijayaraghavan et D.Basu [7]	2020	ANN [b]	0.887309
Na, Jin-Cheon, and Wai Yan Min Kyaing [20]	2015	Linguistic Approach	0.69
Gräber, Felix, et al. [6]	2018	Régression logistique	0.9224
Bemila, et al. [3]	2020	RNN-BiLSTM	0.83906
Y. Mercadier et al. [11]	2020	Roberta avec DAIA	0.316
C. Colón-Ruiz and I. Segura-Bedmar [10]	2020	LSTM	0.9046
M.E BASIRI et al. [21]	2020	3W3DT-NB	0.8836
Our approach	2022	Embedding-100	0.9984

[a] A Generalized Autoregressive Pretraining for Language Understanding (XLNet).
[b] Artificial Neural Network.

4 Conclusion and Future Work

In this work, we propose the Embedding-100 model to predict the drug's effectiveness and efficiency. We rely on natural language processing. As a new approach, we tackle a crucial challenge: the medical semantics to analyze and extract knowledge from medical text. The proposed model is the result of an extensive study. The latter shows that the accuracy of the models depends on their architecture and type.

Furthermore, those models are influenced by the text preprocessing method and sampling type. Experiments indicate that the accuracy (Training accuracy is 0.9984, and the test accuracy is 0.9984) of our model is better than others handled by the totality of the state-of-the-art learning model. Embedding-100 combines artificial neural network, one hot encoder and word embedding. In the future, our work can be extended to analyzing drug effectiveness for other types of data sets, especially medical records. In addition, medical text preprocessing remains a critical research area.

References

1. Haute Autorité de Santé. https://www.has-sante.fr/. Accessed 01 Aug 2022
2. Campesato, O.: Artificial intelligence, machine learning, and deep learning. Mercury Learning and Information (2020)
3. Bemila, T., Kadam, I., Sidana, A., et al.: An approach to sentimental analysis of drug reviews using RNN-BiLSTM model. In: Proceedings of the 3rd International Conference on Advances in Science & Technology (ICAST) (2020)
4. Mascio, A., Kraljevic, Z., Bean, D., et al.: Comparative analysis of text classification approaches in electronic health records. arXiv preprint arXiv:2005.06624 (2020)
5. Mercadier, Y.: Classification automatique de textes par réseaux de neurones profonds: application au domaine de la santé. Diss. Université Montpellier (2020)

6. Graber, F., Kallumadi, S., Malberg, H., et al.: Aspect-based sentiment analysis of drug reviews applying cross-domain and cross-data learning. In: Proceedings of the 2018 International Conference on Digital Health, pp. 121–125 (2018)
7. Vijayaraghavan, S., Basu, D.: Sentiment analysis in drug reviews using supervised machine learning algorithms. arXiv preprint arXiv:2003.11643 (2020)
8. Jiménez-Zafra, S.M., Martín-Valdivia, M.T., et al.: How do we talk about doctors and drugs? Sentiment analysis in forums expressing opinions for medical domain. Artif. Intell. Med. **93**, 50–57 (2018)
9. Yadav, A., Vishwakarma, D.K.: A weighted text representation framework for sentiment analysis of medical drug reviews. In: 2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM), pp. 326–332. IEEE (2020)
10. Colón-Ruiz, C., Segura-Bedmar, I.: Comparing deep learning architectures for sentiment analysis on drug reviews. J. Biomed. Inform. **110**, 103539 (2020)
11. Mercadier, Y., Azé, J., Bringay, S.: Divide to better classify. In: Michalowski, M., Moskovitch, R. (eds.) AIME 2020. LNCS (LNAI), vol. 12299, pp. 89–99. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59137-3_9
12. Min, Z.: Drugs reviews sentiment analysis using weakly supervised model. In: 2019 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), pp. 332–336. IEEE (2019)
13. UCI Machine Learning Repository: Drug Review Dataset. https://archive.ics.uci.edu/ml/datasets/Drug+Review+Dataset+%28Drugs.com%29. Accessed 26 Aug 2022
14. Split Your Dataset With scikit-learn's train_test_split() Real Python. https://realpython.com/train-test-split-python-data/. Accessed 14 Apr 2022
15. Userguide: contents scikit learn. https://scikitlearn.org/stable/user_guide.html. Accessed 10 June 2022
16. Le modèle séquentiel TensorFlow Core. https://www.tensorflow.org/guide/keras/sequential_model. Accessed 12 May 2022
17. Géron, A.: Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media Inc., USA (2019)
18. Kingma, D.P., Jimmy, L.B.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
19. Intuition d'Adam Optimizer – StackLima. https://stacklima.com/intuition-d-adam-optimizer/. Accessed 14 Apr 2022
20. Na, J.C., Kyaing, W.Y.M.: Sentiment analysis of user-generated content on drug review websites. J. Inf. Sci. Theory Pract. **3**(1), 6–23 (2015)
21. Basiri, M.E., Abdar, M., Cifci, M.A., et al.: A novel method for sentiment classification of drug reviews using fusion of deep and machine learning techniques. Knowl.-Based Syst. **198**, 105949 (2020)