Machine Learning for Web Page Classification: A Survey

S.Lassri, H.Benlahmar, A.Tragha
Laboratory of Analysis Modeling and Simulation,
Department of Mathematics and Computer Science,
Faculty of Science Ben M'sik, Hassan II University
Casablanca, Morocco
safae.lassri@gmail.com, h.benlahmer@gmail.com, atragha@yahoo.fr

***Abstract*— The Internet contains a vast amount of data that is growing exponentially. To exploit this data, a Web information retrieval system and a categorization of internet content based on the classification of web pages are essential. Web page classification has many applications, among them the construction of web directories and the building of focused crawlers. In this paper, we present the characteristics of web page classification, we produce a literature review by summarizing and evaluating all sources related to web page classification crawled automatically from ScienceDirect and Springer websites, we review the different machine learning algorithms used to categorize web pages. Finally, we track the underlying assumptions behind the studied methods.**

Index Terms— Machine Learning; Web Mining; Web page classification, Survey

I. INTRODUCTION

The exponential growth of the number of web pages on the internet makes the extraction and the organization of data an impossible task for internet users. Despite the existing search engines, users are confronted with a vast number of suggestions within a keyword search. To solve this problem, web page classification and its applications are used.

Web pages are characterized by their representation in semi-structured documents in HTML, their noisy contents and their links with other web pages by hyperlinks or by query results.

In this paper, we have collected the latest methods to inform future classifier implementations. We provide an overview of the existing machine learning algorithms used for web page classification and a review of the available and known work for each of them. We produce a literature review. We track some assumptions from the studied methods, and we compare them based on some characteristics.

The rest of this article is organized as follows: the background of Web classification and related work are introduced in Section 2; literature review is summarized in section 3; machine learning algorithms used for web page classification are reviewed in section 4, we then point out some interesting directions and conclude the article in Section 5.

II. WEB PAGE CLASSIFICATION

A. Definition of Web Page Classification

Web page classification, also known as a web page categorization, may be defined as the task of determining whether a web page belongs to a category or categories.

Choi and Yao [1] gave a formal definition as bellow: "Let $C = \{c_1, \dots \dots, c_k\}$ be a set of predefined categories, $D = \{d_1, \dots \dots, d_N\}$ be a set of web pages to be classified, and A = D x C be a decision matrix as described in TABLE I.

Where, each entry a_{ij} $(1 \le i \le N, 1 \le j \le K)$ represents whether web page d_i belongs to category c_j or not. Each $a_{ij} \epsilon \{0,1\}$ where 1 indicates web page d_i belongs to category c_j, and 0 for not belonging. A web page can belong to more than one category. The task of web page classification is to approximate the unknown assignment function $f: D \times C \rightarrow \{0,1\}$ using a learned function $f': D \times C \rightarrow \{0,1\}$, called a classifier, a model, or a hypothesis, such that f' coincides to f as much as possible [2].

The function f' is usually obtained by machine learning over a set of training examples of web pages. Each training example is tagged with a category label. The function f' is induced during the training phase and is then used during the classification phase to assign web pages to categories" [1].

TABLE I. Decision Matrix

Web Pages	Categories				
	c_1	…	c_j	…	c_k
d_1	a_{11}	…	a_{1j}	…	a_{1K}
…	…	…	…	…	..
d_i	a_{i1}	…	a_{ij}	…	a_{ik}
…	…	…	…	…	…
d_N	a_{N1}	…	a_{Nj}	…	a_{NK}

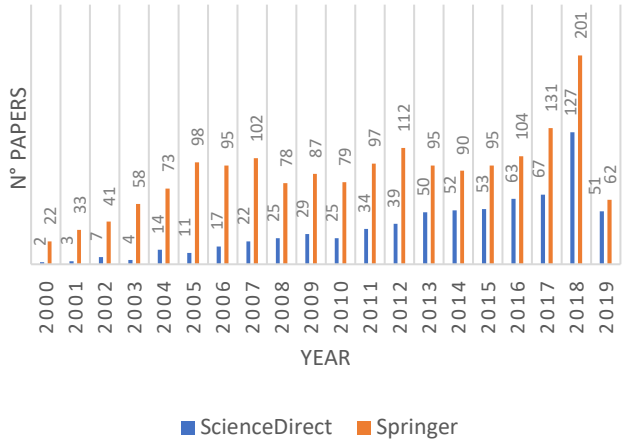


Fig. 1. Year-wise distribution of papers

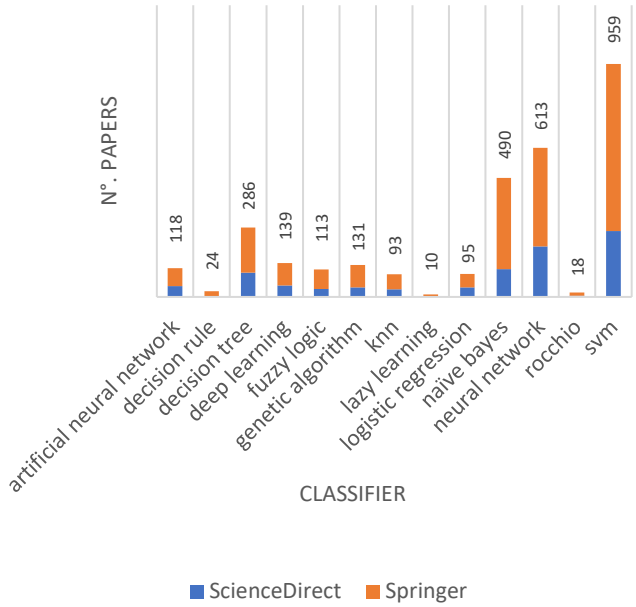


Fig. 2. Distribution of papers for classifiers.

B. Types of Classification

Based on the number of classes available, a classification problem can be divided into a binary classification in which instances should belong to one of two classes, and into a multiclass classification where more than one class is defined.

When only one label is assigned to an instance, the classification problem is defined as single-label classification. But if more than one class is assigned to an instance, the classification is then referred to as multilabel one.

We can also divide web page classification into flat and hierarchical classification where categories are parallel in the former and organized in a hierarchical tree structure in the latter, in which each category may have several subcategories.

C. Applications of Web Page Classification

There are many applications of web page classification, and some of them are web content filtering, ontology annotation, assisted web contextual advertising and knowledge base construction, constructing, maintaining or expanding web directories (web hierarchies), helping question answering systems to improve the quality of search results, building efficient focused crawlers or vertical (domain-specific) search engines, improving quality of search results.

D. Related work

The subject of text classification is well studied in many papers that define all their characteristics and review all relative methods. However, in the case of web page classification, limited review and survey papers are developed. Among them, we can cite the following ones. [1] presents the automatic web page classification systems, and the techniques used to build it. It starts with a definition of web page classification and a description of two types of classifications: subject-based classification and genre-based classification. It then describes how to encode or represent web pages for facilitating machine learning and classification processes. Next, it introduces methods to reduce the dimensionality and discuss the state of the art classifiers in terms of web page classification. Finally, it evaluates many web page classifiers. [3] review work in Web page classification, note the importance of the Web-specific features and algorithms, describe state-of-the-art practices. [4] compare some algorithms used for web page classification employing WEKA clustering/classification algorithms.

III. SYNTHETIC REVIEW OF LITERATURE

To have a background in the proposed studies, we created a synthesis matrix that helps us record the main points of each source and document how sources relate to each other. We generated this matrix automatically by writing a scraping script, which is the process of downloading data from ScienceDirect [5] and Springer [6] websites after introducing web page classification as the search keyword and extracting valuable information from that data. We develop this script with python and beautifulsoup, which allows us to manually extract the elements needed for our study from the selected websites. Each matrix contains that information about each article: year of publication, title, link, type, authors, abstract, keywords, used classifiers, highlights for science direct articles and references for springer articles. We choose springer and ScienceDirect because they contain the largest number of articles related to the topic of web page classification. Post-processing is necessary to make data cleanest.

Fig.1 depicts the number of papers dealing with web page classification within each year. We notice that the web page classification topic has moved from marginalization to mainstream in recent years. It is increasingly treated from 2004 and yields a peak in 2018.

SVM and neural network are the most used classifiers, according to fig.2. They marked a difference of a hundred

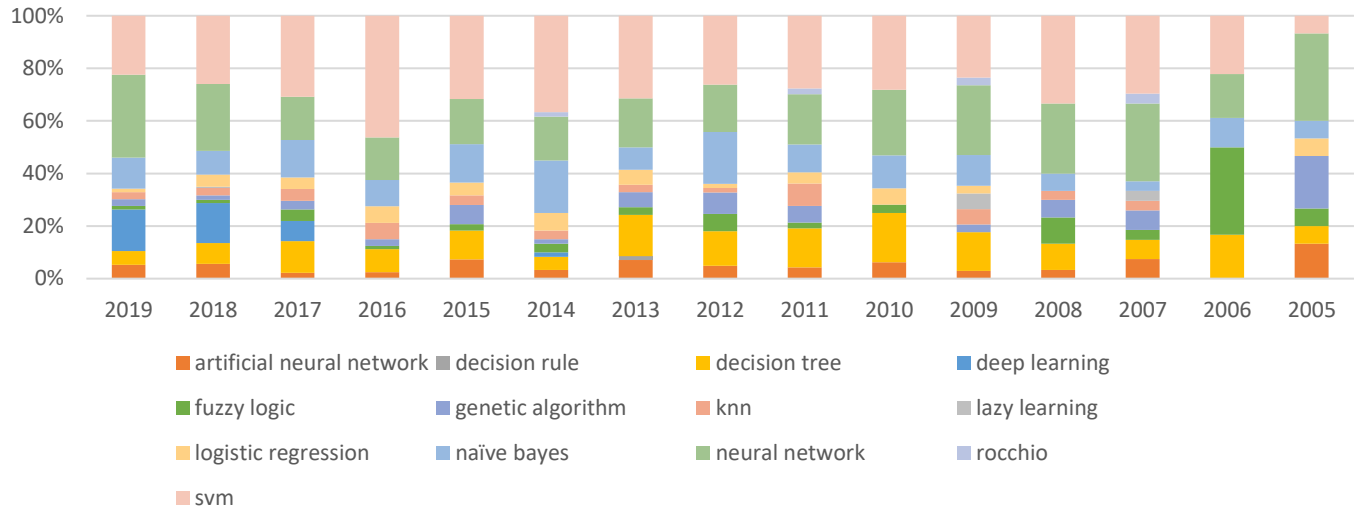


Fig. 3. Yearwise distribution of articles for a classifier.

articles with Naïve Bayes and decision tree.

Fig.3 describes the evolution of the percentage of classifiers used over the years, which reinforces the last remark and shows that SVM and neural network represent more than 50% of the classifiers used. It has also been noted that deep learning is increasingly being chosen in the last three years.

IV. MACHINE LEARNING TECHNIQUES

A. K-Nearest Neighbors

KNN stands for k-Nearest Neighbors. This is one of the simplest techniques to build a classification model. The basic idea is to classify a sample based on its k neighbors. So, samples with similar input values should be labeled with the same target. label. This means that the classification of a sample is dependent on the target labels of the neighboring points.

When multiple neighbors are considered, a voting scheme is used. The majority of the vote is commonly used, so the label associated with the majority of the neighbors is used as the label of the new sample.

With kNN, some measure of similarity is needed to determine how two samples are close together. This is necessary to determine which samples are the nearest neighbors. Distance measures such as Euclidean distance are commonly used. Other distance measures that can be used include Manhattan and hemming distance.

In kNN there is no separate training phase, there is no separate part where a model is constructed, and its parameter is adjusted. This is unlike most other classification algorithms. KNN can generate complex decision boundaries allowing for complex classification decisions to be made. It can be susceptible to noise because classification decisions are made using only information about a few neighboring points instead of the entire dataset. KNN can be slow since the distance between a new sample and all sample points in the data must be calculated to determine the k Nearest Neighbors.

An adaptation of the k-Nearest Neighbor (k-NN) approach is proposed in [7], it is called LIC.To improve its performance, the k-NN approach is supplemented with a feature selection method to reduce noise terms in training samples and a term-weighting scheme using markup tags, and reform document-document similarity measure used in vector space model. In the experiments on a Korean commercial Web directory, the proposed methods in the k-NN approach for Web page classification improved the performance of classification. The Web Page Classification Based on Link Information [8] is the improvement of the KNN algorithm. First, it should be found out which K articles are the most similar to the web pages to be classified at the training collection. Then the new pages' categories should be determined according to the K articles. The main difference between LIC and KNN algorithm is that: the latter classifies relying on web content; the former is dependent on its parent page reference information. The LIC algorithm determines the category attribute of the current page through the links which other web pages point to the current page.

Improvement of results provided by kNN and other text classifiers is proposed [9]. This approach proceeds in four steps: Neighbors Discovery, Classification, Cliques Extraction, and Correction. The contribution of this work is a proposition of a post-classification corrective approach called Clique Based Correction (CBC) that extracts cliques from the implicit graph whose vertices are web pages and edges are implicit links to make categories rectifications for classification results improvement.

LWCS is a fast and low storage usage classification system [10]; it is oriented to large-scale web page classification. Anchor graph hashing is integrated with K- Nearest Neighbors

(kNN) classifier to reduce the pages' original feature dimensions. The original vectors are replaced by the hash value of each page, used during the training and classification phase.

In [11], authors present a similarity computation technique that is based on implicit links extracted from the query-log and used with K-Nearest Neighbors (KNN) in web page classification. The new computed similarity based on clicks frequencies helps enrich KNN for web page classification. This similarity uses neighborhood information and helps reduce the effect of the problem of dimensionality faced when using KNN based on the text-only. To classify a web page p_i, KNN calculates similarities between p_i and each web page in the training set. Then, it ranks web pages in the training set based on those similarities. In our case, KNN does a two-level ranking. First, it ranks web pages using the implicit links-based similarity. Then, web pages having this similarity equal to zero are ranked again using the cosine similarity.

B. Support Vector Machine

Support vector machines (SVMs) are one of the discriminative classification methods which are commonly recognized to be more accurate. The SVM classification method is based on the Structural Risk Minimization principle from computational learning theory [12]. The idea of this principle is to find a hypothesis to guarantee the lowest true error.

The SVM need both positive and negative training set. These positive and negative training sets are needed for the SVM to seek the decision surface that best separates the positive from the negative data in the n-dimensional space, so-called the hyperplane.

Furthermore, it can handle documents with high-dimensional input space and culls out most of the irrelevant features [13]. However, the major drawback of the SVM is their relatively complex training and categorizing algorithms and also the high time and memory consumptions during the training stage and classifying stage.

Additionally, Support vector machines (SVM) offers one of the most robust and accurate methods among all well-known algorithms [14]. It has a sound theoretical foundation, requires only a dozen examples for training, and is insensitive to the number of dimensions. Besides, efficient methods for training SVM are also being developed at a fast pace.

The effect of using context features in web classification using SVM classifiers is studied in [15]. The use of the title components and anchor words as context features improved the classification accuracy significantly. Compared with Foil-Pilfs, the experiments show that SVM-based web classification methods performed very well on the WebKB data set.

The classification of web pages using only a well-treated URL by SVM machine learners exceeds the performance of some source document-based features[16]. URL is treated in a two-stage process. The Information content reduction uses information content as a criterion for splitting, and title token-based finite-state transducer (FST) tries to simultaneously split and expand segments based on previously-seen web page titles.

The authors of [9] demonstrate that the Clique Based Correction approach improves the results of the SVM classifier as well as the kNN classifier.

In [17], the authors proposed an implicit link-based Gaussian kernel that uses an implicit links-based distance. This kernel helps enrich SVM for web page classification by involving users' intuitive judgments in the classification. Results show that implicit links-based kernel helps to bring improvements on SVM's results.

Using multi-LDA instead of CHI for feature extraction yield a higher precision and improve the results of the SVM classifier [18]. Multi- LDA extracts features considering semantic characteristics.

C. Naïve Bayes

A Naïve Bayes classification model [19] uses a probabilistic approach to classification. The class with the highest probability then determines the label for the sample. In addition to using a probabilistic framework for classification, the Naïve Bayes classifier also uses what is known as Bayes' theorem. Naïve Bayes assumes that the input features are statistically independent of one another. This means that, for a given class, the value of one feature does not affect the value of any other feature. This independence assumption is an oversimplified one that does not always hold. The naïve independence assumption and the use of Bayes theorem give this classification model its name.

The classification task is defined as follows: Capital X is the set of values for the input features in the sample, given a sample with features X, predict the corresponding class C. So, for classification, we want to find the value of C that maximizes the probability of C given X. Using Bayes' theorem, the probability of c given x can be expressed using other probability quantities, which can be estimated from the data:

$$P(C|X) = \frac{P(X|C) * P(C)}{P(X)} \qquad (1)$$

So, for classification the posterior probability P(C|X) should be calculated for each class C, and $P(C)$ should be estimated by the calculation of the fraction of samples for class C in training data. Moreover, to estimate $P(X|C)$, only need to estimate $P(X_i|C)$ individually.

The Naïve Bayes classification model is a fast and simple algorithm and the probabilities that are needed can be calculated with a single scan of the data set and stored in a table. However, the major drawback of the Naïve Bayes classification model is that the independence assumption does not hold true in many cases. The independence assumption also prevents the naïve base classifier to model interactions between features which limits its classification power.

In [20], text documents are represented by a vector of feature which includes up to five consecutive words and then classified

by a Naïve Bayesian classifier. This n-gram representation can capture the concept of phrases, which are unlikely to be characterized using single terms. Therefore, it generates a space with much higher dimensionality.

The use of Naïve Bayes in hypertext categorization using hyperlinks is shown in [21]. A set of adjacent documents is generated by keeping only the neighbors that are homogeneous with the target document. For all terms in the target document, the term weight is adjusted using the term frequencies in the neighbor documents, so that the content of the target document is influenced by the contents of the neighbors. A temporary class is assigned to all neighbors that have not been categorized, after a partial confidence value is assigned to those having a temporary label. Finally, the probability value for each class is calculated, and the best class is chosen.

The high dimensionality issue is the major problem in web page classification [22]. Naïve Bayesian classifier provides good results with dimensionality reduction methods. A hybrid approach of dimensionality reduction for web page classification is described using a rough set Quick Reduct algorithm for dimensionality reduction and information gain as a feature selection method.

In [23], a web page classification technique is proposed to mine news articles from a web corpus. It uses the content of the web page, web page URL and structure information on a web page to identify the news pages from non-news pages. In this case, the Naïve Bayes classifier performs better than SMO and J48.

In this paper [11], authors introduce an implicit links-based probability computation method used with Naive Bayes (NB) for web page classification. The new computed probability using frequencies of clicks help enrich NB for web page classification. Those frequencies involve users' intuitive judgments and neighboring information in probabilities computation. Authors modify the probability computation mechanism of Naive Bayes (NB) to use weights of edges relating web pages and involve users' intuitive judgments in the process of probabilities computation. Instead of using solely the probability of a category c given a web page p, authors employ the probability of c given a web page p and its neighbors.

D. Artificial Neural Network

An Artificial Neural Network (ANN), often just called a "Neural Network" (NN), is a mathematical model or computational model based on biological neural networks [24]. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. In most cases, an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase.

The word network in the term 'artificial neural network' arises because the function $f(x)$ is defined as a composition of other functions $g_i(x)$, which can further be defined as a composition of other functions. This can be conveniently represented as a network structure, with arrows depicting the dependencies between variables. A widely used type of composition is the nonlinear weighted sum.

$$f(x) = K(\sum_i w_i\, g_i(x)) \qquad (2)$$

Where K is some predefined function, such as the hyperbolic tangent. It will be convenient for the following to refer to a collection of functions g_i as simply a vector $g = (g_1, g_2, \dots\dots, g_n)$.

The news web page classification method (WPCM) [25] uses a neural network with inputs obtained by both the principal components (PCA) and class profile-based features (CPBF). Each news web page is represented by the term-weighting scheme. The principal component analysis has been used to select the most relevant features for the classification. Then the final output of the PCA is combined with the feature vectors from the class-profile, which contains the most regular words in each class. Experiments showed that WPCM increases the classification accuracy compared with TF-IDF, Bayesian and PCA-NN classification methods.

Another approach for web page categorization using a hybrid neural network is proposed [26]. A web page is represented by a vector of features with different weights according to the term frequency and the importance of each sentence on the page. As the number of features is big, PCA is used to select the relevant features. Finally, the output of PCA is sent to SOFM for classification. This method makes a significant improvement in classifications compared with k-NN and Naïve Bayes.

In this work [27], they aimed to develop a classifier that can categorize web pages based on their ability to attract random surfers. Web pages are classified into "bad" and "not bad" classes, where the "bad" class implies poor attention drawing ability. In the proposed approach, the web page content is divided into objects. The area occupied by these objects served as the attribute of the classifier. The experiments with various classification algorithms supported by the WEKA tool prove that two of those, namely the random subspace and the RBF networks, gives high accuracy (83.33%) with high precision and recall.

The authors in [28] proposed a method for classifying the E-commerce web pages using MLP neural network, automatic content extraction, and automatic keyword extraction. First of all, the SDND method is used to detect the subject of the web page and its content. Then the text preparation technique is used to transform the text into the extracted content into words. Next, MRF's method is applied to select some related keywords to generate feature vectors. Finally, The E-commerce web pages are classified using the MLP neural network. Compared with Naive Bay, RBF, and SVM, this method can achieve the highest accuracy of 97.60 percent.

The aim of [29] is to propose a novel neural network model with high accuracy and good generalization ability for detecting phishing websites. Different from the traditional neural network, this novel neural network classification method

contains the design risk minimization principle and Monte Carlo algorithm. This novel neural network classification method can avoid over-fitting by design risk minimization principle.

E. Deep learning

The main challenge of Artificial Intelligence is the ability of a machine to represent and model the data and information in a way our human brain does. Unlike traditional classification methods, which require handcrafted features to be extracted and pre-processed so that the classification techniques can be applied, deep learning techniques learn features while going through multiple hidden layers. The deep network learns high-level abstract features progressively. This is possible due to the passing information learned in the previous layers to the future layers.

Unlike a deep network, a normal neural network cannot reason with the events that have occurred in the past, as each computation layer of this type of network is independent and does not affect each other. Thus, they are ‘‘stateless" and cannot learn the information from the past sequences which are their major drawback.

In [30] a spam classification is implemented by a special architecture of deep learning technique known as Long Short Term Memory (LSTM) a variant of the Recursive Neural Network (RNN). Unlike traditional classifiers, LSTM can learn abstract features. Before using the LSTM for the classification task, the text is converted into semantic word vectors with the help of word2vec, WordNet and ConceptNet. The evaluation of the results shows that LSTM can outperform traditional machine learning methods for the detection of spam with a considerable margin.

The authors of [31] present a deep learning method using Stacked denoising autoencoders model to learn and detect intrinsic malicious features. They employ a SdA network to analyze URLs and extract features automatically. Then a logistic regression is implemented to detect malicious and benign URLs, which can generate detection models without a manually feature engineering. This architecture outperforms other deep learning models and feature-engineer models.

A novel framework for the categorization of web pages based on their visual content is proposed in [32]. This is achieved by exploring the joint application of a transfer learning strategy and metric learning techniques to build a Deep Convolutional Neural Network (DCNN) for feature extraction, even when training data is scarce. The obtained experimental results evidence that the proposed approach outperforms the state-of-the-art handcrafted image descriptors and achieves a high categorization accuracy. Also, the problem of over-time learning is addressed, so the proposed framework can learn to identify new web page categories as new labeled images are provided at test time. As a result, prior knowledge of the complete set of possible web categories is not necessary for the initial training phase. This paper builds upon the ideas and results presented in [33], where the authors explored the applicability of deep learning techniques to the problem of web page classification by adopting a transfer learning strategy.

In this study, fake web sites were identified using deep learning. In the classification process, feedforward Neural networks and stacked automatic encoders are used. To detect fraudulent websites, URLs belonging to websites that are punctuated by the internet are collected and analyzed together with malicious websites. Compared with SVM and decision tree, this method can achieve the highest accuracy of 86 percent.

F. Decision Tree

The idea behind decision trees [34] for classification is to split the data into pure regions, that is regions with samples from only one class. With real data, completely pure subsets may not be possible. So, the goal is to divide the data into subsets that are as pure as possible. That is each subset contains as many samples as possible from a single class. Boundaries separating these regions are called decision boundaries. And the decision tree model makes classification decisions based on these decision boundaries.

The algorithm for constructing a decision tree model is referred to as an induction algorithm. At each split the induction algorithm only considers the best way to split that particular portion of the data. This is referred to as a greedy approach. In the Greedy algorithms, the tree has to be built in piecemeal fashion by determining the best way to split the current node at each step, and combining these decisions to form the final decision tree.

It turns out that it works out better mathematically if the impurity is measured rather than the purity of a split to compare different ways to partition a set of data. So, the impurity measure of a node specifies how mixed the resulting subsets are. A common impurity measure used for determining the best split is the Gini Index. The lower the Gini Index, the higher the purity of the split. Besides the Gini Index, other impurity measures include entropy, or information gain, and misclassification rate. The other factor in determining the best way to partition a node is which variable to split on. The decision tree will test all variables to determine the best way to split a node using a purity measure such as the Gini index to compare the various possibilities.

One of the advantages of decision trees for classification is that the resulting tree is often simple to understand and interpret. Another one is that the tree induction algorithm is relatively computationally inexpensive, so training a decision tree for classification can be relatively fast. The greedy approach used by the tree induction algorithm determines the best way to split the portion of the data at a node but does not guarantee the best solution overall for the entire data set. Decision boundaries are rectilinear. This can limit the expressiveness of the resulting model which means that it may not be able to solve complicated classification problems that require more complex decision boundaries to be formed.

Feature selection and discretization are the major preprocessing done before induction. The feature selection is

made by Cfs subset evaluator, then the final set of features is selected using a decision tree-based classifier, C4.5. In this paper [35], they implemented an algorithm which discretizes the features for web page classification. The results have shown a good improvement in classification accuracy with discretized features than with continuous features.

Various classification algorithms based on decision tree are evaluated and analyzed to separate non-advertisement and advertisement websites[36]. The results showed that J48 outperform Decision Stump, Hoeffding tree, Logistic model tree (LMT), Random Forest algorithm, Random tree, REP Tree classifiers.

A web spam detection system is proposed in [37]. It combines the clonal selection algorithm for feature selection and under-sampling for data balancing. The system builds several C4.5 decision sub-classifiers from the balanced datasets based on its specified features. Finally, these sub-classifiers are used to construct an ensemble decision tree classifier, which is applied to classify the examples in the testing data.

In [38], authors present a method for detecting hijacked journals by using a classification algorithm. For this purpose, they use a dataset related to hijacked and authentic journals for using a classification algorithm and preparing a decision tree. Nine features are used for detecting hijacked journals. Four of them were adopted from previous studies [39] [40]. They include domain rank in the search engine, age of the domain, entering countries in the journal website, and aim and scope of the journal. Five features are new. They include the number of broken links, the number of the published articles in a year, consistency between the country of the server and the country of the journal, number of dead links and use of the character ‘‘-" in the URL. Algorithms Decision Stumps, J48, Random Tree, and REP Tree were applied to the training dataset in WEKA. Next, the algorithm with the lowest error rate in creating classes was selected as the best algorithm for identifying hijacked journals. As a result, Random Tree had the lowest error rate.

A distance-based decision tree learning algorithm (DBDT) is proposed in [41]. It allows decision trees to handle structured attributes (lists, graphs, sets, etc.) along with the well-known nominal and numerical attributes. These structured attributes are employed to represent the content and structure of the website. This algorithm differs from traditional ones in the sense that the splitting criterion is defined using metric conditions (“is nearer than”).

G. Hybrid classifiers:

A hybrid classifier combines several machine learning techniques to improve system performance. More specifically, a hybrid approach typically consists of two functional components. The first one takes raw data as input and generates intermediate results. The second one will then take the intermediate results as the input and produce the final results [42].

In particular, hybrid classifiers can be based on cascading different classifiers. On the other hand, hybrid classifiers can use some clustering-based approach to preprocess the input samples to eliminate unrepresentative training examples from each class. Then, the clustering results are used as training examples for classifier design. Therefore, the first level of hybrid classifiers can be based on either supervised or unsupervised learning techniques. Finally, hybrid classifiers can also be based on the integration of two different techniques in which the first one aims at optimizing the learning performance (i.e. parameter tuning) of the second model for prediction [43].

New feeds are allocated into fixed sections of news like a business, sports [44]. This is done by adopting the hybrid technique of URL analysis and content context analysis. The proposed model which starts with web crawling of URLs, scraping of news contents followed by the analysis carried out on account of generating keywords, weight calculation, and then, at last, identify the relevant category based on contents fetched among various Indian news web portal.

Authors of [45] develop an anti-phishing scheme to tackle phish web pages and mitigate their consequences. This paper proposed a new Phishing Hybrid Feature-Based Classifier (PHFBC) which hybridized two machine learning algorithms (Naïve Base) and (Decision Tree) with a statistical criterion of Phish Ratio. In conjunction, a Recursive Feature Subset Selection Algorithm (RFSSA) was also proposed to characterize phishing holistically with a robust selected subset of features. Outcomes of performance assessment via simulations, real-time validation, and comparative analysis demonstrated that PHFBC was highly distinctive among its competitors in terms of classification accuracy and minimal misclassification of novel phishes on the Web.

In [46], decision tree and ARTMAP approaches are used together with the proposed DTA approach to identify the languages belonging to the Arabic script web documents (Arabic, Persian, Urdu). In the DTA approach, the rule of the DT has been used to extract the features from each document. Then, the ARTMAP has been used to do a language identification process on those input patterns. This combination has improved the performance of the Arabic script language identification on web documents in a variety of languages. The result shows that the proposed approach has outperformed both the decision tree and the default ARTMAP approaches. This method might not function properly in other web domains such as those dealing with biology or chemistry, most of which contain specific characters or terminological terms.

A. Markov, M. Last, and A. Kandel present three hybrid methods of web document representation. These methods are based on frequent sub-graph extraction that can help to overcome the problems of traditional bag-of-words [47] and graph [48] techniques. They evaluate the hybrid representation methodology using two model-based classifiers (C4.5 decision-tree algorithm and probabilistic Naïve Bayes) and two benchmark web document collections. The hybrid model has succeeded in improving the performance of model-based

document classifiers in terms of classification time while preserving nearly the same level of classification accuracy.

H. Ensemble classifiers:

Ensemble classifiers were proposed to improve the classification performance of a single classifier [49]. The term ‘‘ensemble” refers to the combination of multiple weak learning algorithms or weak learners. The weak learners are trained on different training samples so that the overall performance can be effectively improved.

Among the strategies for combining weak learners, the ‘‘majority vote” is arguably the most commonly used one in the literature [43]. Other combination methods, such as boosting and bagging, are based on training data resampling and then taking a majority vote of the resulting weak learners.

Advanced Persistent Threat (APT) is a threat that cannot be predicted via conventional cybersecurity measures as it employs a series of social engineering techniques that tricks the user into providing the credentials to access the system. This project [50] covers the prediction of the possibility of an APT from mobile devices. A Mobile application is developed to obtain and send SMS content to the server for processing. If there is a URL contained in the SMS, APTGuard will extract the feature of the URL and then classify it accordingly using ensemble learner which combines decision tree and neural network accurately.

The aim of [51] is to build a novel ensemble decision tree classifier based on under-sampling (US) and clonal selection (CS) to improve the performance of web spam detection. First, the system will convert the imbalanced training dataset into several balanced datasets using the under-sampling method. Second, the system will automatically select several optimal feature subsets for each sub-classifier using a customized clonal selection algorithm. Third, the system will build several C4.5 decision tree sub-classifiers from these balanced datasets based on its specified features. Finally, these sub-classifiers will be used to construct an ensemble decision tree classifier, which will be applied to classify the examples in the testing data. After comparing and analyzing the accuracy, F1-Measure, and ROC AUC results, the authors conclude that the USCS ensemble classifier outperforms the other traditional classification models.

In this research [52], three best models named Bagged CART, eXtreme Boosting Technique, and Parallel Random Forest out of 15 different classification models have been utilized for the Ensemble approach to classifying spam and non-spam web pages. Then the Fold Cross-validation approach is also used for testing the system, and it also reduces the problem of overfitting. The dataset is shuffled ten times, and the results are cross-validated.

Fayrouz Elsalmy, Rasha Ismail, and Walid AbdelMoez suggested an approach to improve the predictive power of the web page classification models by stacking ensemble method [53]. Random forest, stacking with multi-response model trees and four different base learners (Naïve Bayes, J4.8, IBK, and FURIA) are used. Their results show that stacking with multi-response model trees outperforms random forest and the other existing ensemble methods examined in previous studies.

V. COMPARISON OF WEB PAGE CLASSIFICATION APPROACHES

After comparing all the studied methods, we notice that:

- The preprocessing of web pages is a very important stage that improves considerably the results of machine learning classifiers and decreases the noisy elements on the web pages.
- The exploitation of both types of hyperlinks, implicit and explicit one, increases the classification accuracy and enriches the content of the target web page.
- Neural networks work better than other methods, even when the data contains noise and has a poorly understood structure and changing characteristics.
- Existing algorithms work well with a small number of web pages, whereas they become slow and even noneffective while dealing with a large scale of web pages.
- Deep learning is increasingly chosen in the last three years. The advantage of the deep network is its capability of learning high-level abstract features progressively. This is possible due to the passing information learned in the previous layers to the future layers.

Table II presents a comparative study that mentions some characteristics of the existing methods. Based on this comparative study, we find that the representation of documents in a bag of word format is the most chosen [54]. Additionally, according to the reported results [28], MLP Neural Network achieved the highest accuracy 97,6 percent compared with Naïve Bayes, RBF, and SVM, which are evaluated on the same dataset and with the same features representation.

TABLE II. Comparison of Web Page Classification Approaches

Year	Approach	Classifier	reported results	Document representation	feature selection criteria	evaluation dataset
2019	APTGuard: Advanced Persistent Threat (APT) Detections and Predictions using Android Smartphone	ensemble learning that combines decision	sensitivity rate of 91.00%, a fall-out rate of 1.05%, precision of 98.35%, and accuracy of 95.71%.	feature vectors	manually	499 from crazyurl, 412 from PhishTank, 500 from Alexa and the remaining from SMSs

		tree and neural network				
2019	Optimizing semantic LSTM for spam detection	Long Short Term Memory (LSTM)	LSTM outperforms the other traditional classification models when applied to : SMS Spam Dataset: precision:98,74 /Recall:99,35/Accuracy: 99,01/F1: 99,24 Twitter Dataset: precision:95,54 /Recall:98,37/Accuracy: 95,09/F1: 96,84	semantic word vectors	most frequently occurring words	benchmark SMS spam dataset, available in UCI repository and Tweets dataset, extracted from public live tweets from the microblogging site Twitter using Twitter API
2019	Detecting Malicious URLs Using a Deep Learning Approach Based on Stacked Denoising Autoencoder	SdA-logistic regression model	accuracy of 98.25% and a micro-averaged F1 score of 0.98 much higher than the other deep learning model and feature-engineer model (SdA-SVM, SVM, LSTM, Bayes)	vector of the encoded URL	Autoencoders	1 million URLs crawled from Alexa top 1 million domain websites + 1 million URLs came from Common Crawl+ 1 million phishing URLs came from Phishtank+ 0.5 million malicious URLs came from Anti Network-Virus Alliance of china+ 0.5 million malicious URLs came from hpHosts
2019	Visual content-based web page categorization with deep transfer learning and metric learning	Deep Convolutional Neural Network	91.85% accuracy for individual image categorization and a 98.97% accuracy for web page categorization	high-level feature descriptor for each of the extracted images from the visual content of the web page	The feature extraction module, based on a DCNN	An extended dataset consists of a total of 365 web pages distributed. These web pages contained a total of 4027 images
2019	Phishing Analysis of Websites Using Classification Techniques	stacked autoencoders	Accuracy equal to 86% with an improvement of 26% and 5% of SVM and decision tree	matrices containing ASCII values of encoded url	An autoencoder extracts features from their input in two-phase: encoding and decoding	1000 URLs of web sites belonging to non-malicious web sites which are collected from authors' browsing history +1000 malicious URLs from PhisTank website
2018	Phishing Hybrid Feature-Based Classifier by Using Recursive Features Subset Selection and Machine Learning Algorithms	Hybrid: Naïve Base and Decision Tree with a statistical criterion of Phish Ratio	achieved (97%), (0.7%), (0%), and (98.07%) average rates of True Positive Rate, False Positive Rate, False Negative Rate, and AUC respectively.	Feature Vector (58 features included ten URL features, 24 Cross-Site Scripting (XSS) features, and 24 HTML features)	Recursive Feature Subset Selection Algorithm (RFSSA)	N/A
2018	The application of a novel neural network in the detection of phishing websites	neural network	High accuracy of 97.71% and a low FPR of 1.7%.	30 features	N/A	From the UCI repository. This dataset collects mainly from PhishTank archive, MillerSmiles archive, and Google's searching operators.
2017	A novel ensemble decision tree based on under-sampling and clonal selection for web spam detection	C4.5	the USCS classifier outperforms the other traditional classification models with F1= 0,92	N/A	clonal selection algorithm	WEBSPAM-UK2006
2017	Spammer Classification Using Ensemble Methods over Content-Based Features	Bagged CART,eXtreme_Gradient_Boosting and	the accuracy is raised to 89.17, Cross-validation Validation set approach has resulted in 90.39% accuracy in 10 rounds	N/A	random-forest machine learning approach	UK-2011 webspam dataset

		Parallel Random Forest				
2017	A novel ensemble decision tree based on under-sampling and clonal selection for web spam detection	novel ensemble decision tree based on under-sampling and clonal selection	Accuracy equal to 89.68% with an improvement of 6% and 14,5% and 12% and 17% and 20% of (C4.5+US) and (C4.5+ Adaboost) and (C4.5 + bagging) and C4.5 and NaiveBayes	feature vectors	customized clonal selection algorithm	WEBSPAM-UK2006 and WEBSPAM-UK2007
2017	Detecting Hijacked Journals by Using Classification Algorithms	Decision tree (Random Tree)	10% error rate	Feature Vector	N/A	A dataset composed of 104 data records from journal websites: 59 were authentic and 45 were hijacked, 45 were hijacked.
2016	News web page classification using URL content and structure attributes	Naïve Bayes algorithm	precision greater than 0,9 and Naïve Bayes classifier perform better than SMO and J48	vectors of features (structure, URL, content attributes)	N/A	Indian news websites corpus
2016	Towards effective web page classification	SVM	Accuracy equal to 84.15%, 2.23% superior to the traditional classification strategy based on CHI	Bag of words	Multi- LDA	Sogou corpus
2016	Implicit Links-Based Techniques to Enrich K-Nearest Neighbors and Naive Bayes Algorithms for Web Page Classification	KNN and NB	Improvement of 1.2% of micro F1 and macro F1 averages given by KNN using implicit links-based similarity compared to text-based similarity	web pages' implicit graph	AOL query-log	Open Directory Project (ODP)
2016	Enhancing Web Page Classification Models	Random forest, stacking with multi-response model trees and four different base learners (Naïve Bayes, J4.8, IBK and FURIA)	91.2 % classification accuracy with an improvement of 4% and 13,5% and 25,5% and 13% of naïveBbayes and c4.5 and Ib1 and FURIA classifier	N/A	N/A	50 datasets of DMOZ (Open Directory Project)
2015	Implicit Links Based Kernel to Enrich Support Vector Machine for Web Page Classification	SVM	N/A	Bag of words	N/A	Open Directory Project (ODP)
2015	Hybrid Dimensionality Reduction Approach for Web Page Classification	Naïve Bayesian	micro averaging measure is greater than 0.9, 10% of improvement in accuracy with feature selection	Bag of words	Information gain method	CSMINING GROUP (webkb+ r8+20ng)
2015	LWCS: A Large-scale Web Page Classification System Based on Anchor Graph Hashing	KNN	10 times faster than the original one. Nearly the same accuracy	Hash value	term frequency-inverse document frequency (tf-idf) and anchor graph hashing	N/A
2015	E-commerce web page classification based on automatic content extraction	MLP neural network	Compared with Naive Bayes, RBF, and SVM, this method can achieve the highest accuracy of 97.60 percent	feature vectors with generated kye words of the web page	SDND method and MRFs method	collected from 3 online store web sites including Jabong, Global Reebok, and Asos.

2015	Evaluation and analysis of popular Decision tree algorithms for annoying advertisement websites classification	Decision tree classifier	J48 decision tree has the highest classification accuracy among other algorithms with 71.4286%.	ARFF format	N/A	“Bing” search engine
2014	A Clique Based Web Page Classification Corrective Approach	SVM, NB or KNN	N/A	N/A	N/A	Open Directory Project (ODP)
2012	Classification of web pages on attractiveness: A supervised learning approach	IB1(Nearest-neighbors classifier) AND Random subspace and RBF network	algorithms RBF network and Random subspace gives the best performance, with about 83% accuracy	divide the web page content into six objects	N/A	30 web pages from the web site that lists the poorly designed web pages(www.webpagesthatsucks.com) and 30 web pages crawled randomly from the web
2012	A supervised discretization algorithm for web page classification	decision tree-based (J48 and ID3)	J48 and ID3 have shown a significant improvement in accuracy with discrete features	Discretized web page Feature Vectors	Cfs and discretization algorithm	WebKB
2011	Arabic script web page language identifications using decision tree neural networks	hybrid decision tree-ARTMAP	Accuracy equal to 99.49% with an improvement of 18% and 27% of decision tree and ARTMAP	Input vector (decision tree)	letter frequency	news data set collected from the BBC website
2011	A Web Page Classification Algorithm Based On Link Information	kNN	N/A	N/A	Document frequency	integrated portal Sohu, Netease, Yahoo and professional website javaeye, csdn
2007	Fast Categorization of Web Documents Represented by Graphs	C4.5 decision-tree algorithm and probabilistic Naïve Bayes	best classification accuracy with a hybrid method an da significant increase in the categorization speed (the shortest total time is reached with Hybrid Smart using the fixed threshold approach, where the highest accuracy is also reached)	graph	Hybrid Naïve, Hybrid Smart, and Hybrid Smart with Fixed Threshold	K-series [55]and the U-series [56]
2005	A Novel Framework for Web Page Classification Using Two-Stage Neural Network	Neural network	F1 equal to 86,87% with an improvement of 2% and 5% of KNN and Naïve Bayes	term-weighting vector	PCA	Data set of sports news obtained from the Yahoo.com and Google.com
2004	Web page categorization without the web page	SVM	N/A	URL	finite state transducer (FST)	WebKB corpus
2004	Web page feature selection and classification using neural networks	Neural network	F1 equal to 91,65% for the WPCMhhwich perform better than PCA-NN and TF-IDF	term-weighting scheme	the principal components analysis (PCA) and class profile-based features (CPBF)	sports news web obtained from Yahoo server
2002	Using Web structure for classifying and describing Web pages	SVM	more than 98% on average for negative documents, and as high as 96% for positive documents, with an average of about 90%	Bag of words	entropy-based dimensionality reduction	Yahoo and WebKB
2002	Web classification using support vector machine	SVM	0.6 values for the F1 measure	Text + Title + Anchor Words	N/A	WebKB
2000	Web page classification based oa k-nearest neighbor approach	KNN	From 18,2% to 19,2% (micro averaging breakevan point)	Bag of words	Expected mutual information (EMI) and mutual information (MI)	Hanmir

2000	A Practical Hypertext Categorization Method using Links an Incrementally Available Class Information	Naïve Bayesian	18.5% of improvement in effectiveness, the increase of -Fscore was by 6,7%	N/A	N/A	Reuter-21578 collection and ETRI-Kyemong
1998	Turning Yahoo into an Automatic Web-Page Classifier	Naïve Bayesian	N/A	N-gram	Gram frequency	yahoo directory

VI. CONCLUSION

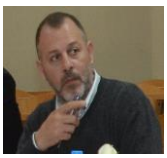
In the case of web page classification, we map each web page to one category or multiple categories. This classification plays an important role in data extraction systems as well as search engines, contextua webl advertising and others. The phase of features extraction and reduction is critical because of its impact on classifier's accuracy, as well as the choice of the classifier. In this wor,k we reviewed the existing machine learning algorithms used for web page classification, we produced a literature revie,w and we compared related methods based on some characteristics. For future work, the visual analysis of web pages, the removal of the noisy content and the implicit and explicit links with other pages should be taken into consideration, to have the maximum accuracy possible.

REFERENCES

[1] B. Choi et Z. Yao, « Web page classification », in Foundations and Advances in Data Mining, Springer, 2005, p. 221–274.
[2] F. Sebastiani, « Machine learning in automated text categorization », ACM computing surveys (CSUR), vol. 34, no 1, p. 1–47, 2002.
[3] X. Qi et B. D. Davison, « Web page classification: Features and algorithms », ACM computing surveys (CSUR), vol. 41, no 2, p. 12, 2009.
[4] I. Charalampopoulos et I. Anagnostopoulos, « A comparable study employing weka clustering/classification algorithms for web page classification », in Informatics (PCI), 2011 15th Panhellenic Conference on, 2011, p. 235–239.
[5] « 36 798 Search Results - Keywords(web page classification) - ScienceDirect ». [En ligne]. Disponible sur: https://www.sciencedirect.com/search?qs=web%20page%20classification&show=25&sortBy=relevance.
[6] « Search Results - Springer ». [En ligne]. Disponible sur: https://link.springer.com/search?query=web+page+classification.
[7] O.-W. Kwon et J.-H. Lee, « Web page classification based on k-nearest neighbor approach », in Proceedings of the fifth international workshop on Information retrieval with Asian languages, 2000, p. 9–15.
[8] Z. Xu, F. Yan, J. Qin, et H. Zhu, « A web page classification algorithm based on link information », in Distributed Computing and Applications to Business, Engineering and Science (DCABES), 2011 Tenth International Symposium on, 2011, p. 82–86.
[9] B. Abdelbadie et B. Mohammed, « A Clique Based Web Page Classification Corrective Approach », in Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014 IEEE/WIC/ACM International Joint Conferences on, 2014, vol. 2, p. 467–473.
[10] Y. Zheng, C. Sun, C. Zhu, X. Lan, X. Fu, et W. Han, « LWCS: A large-scale web page classification system based on anchor graph hashing », in Software Engineering and Service Science (ICSESS), 2015 6th IEEE International Conference on, 2015, p. 90–94.
[11] A. Belmouhcine et M. Benkhalifa, « Implicit Links-Based Techniques to Enrich K-Nearest Neighbors and Naive Bayes Algorithms for Web Page Classification », in Proceedings of the 9th International Conference on Computer Recognition Systems CORES 2015, vol. 403, R. Burduk, K. Jackowski, M. Kurzyński, M. Woźniak, et A. Żołnierek, Éd. Cham: Springer International Publishing, 2016, p. 755-766.
[12] « The Nature of Statistical Learning Theory | Vladimir N. Vapnik | Springer ». [En ligne]. Disponible sur: https://www.springer.com/la/book/9781475724400. [Consulté le: 26-avr-2018].
[13] A. Khan, B. Baharudin, L. H. Lee, et K. Khan, « A review of machine learning algorithms for text-documents classification », Journal of advances in information technology, vol. 1, no 1, p. 4–20, 2010.
[14] X. Wu et al., « Top 10 algorithms in data mining », Knowledge and information systems, vol. 14, no 1, p. 1–37, 2008.
[15] A. Sun, E.-P. Lim, et W.-K. Ng, « Web classification using support vector machine », in Proceedings of the 4th international workshop on Web information and data management, 2002, p. 96–99.
[16] M.-Y. Kan, « Web page classification without the web page », in Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters, 2004, p. 262–263.
[17] A. Belmouhcine et M. Benkhalifa, « Implicit links based kernel to enrich Support Vector Machine for web page classification », in Intelligent Systems: Theories and Applications (SITA), 2015 10th International Conference on, 2015, p. 1–4.
[18] M. Gu, F. Zhu, Q. Guo, Y. Gu, J. Zhou, et W. Qu, « Towards effective web page classification », in Behavioral, Economic and Socio-cultural Computing (BESC), 2016 International Conference on, 2016, p. 1–2.
[19] T. M. Mitchell, Machine Learning. New York: McGraw-Hill, 1997.
[20] D. Mladenic, « Turning yahoo into an automatic web-page classifier », 1998.
[21] H.-J. Oh, S. H. Myaeng, et M.-H. Lee, « A practical hypertext catrgorization method using links and incrementally available class information », in Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, 2000, p. 264–271.
[22] S. Sarode et J. Gadge, « Hybrid dimensionality reduction approach for web page classification », in Communication, Information & Computing Technology (ICCICT), 2015 International Conference on, 2015, p. 1–6.
[23] C. Arya et S. K. Dwivedi, « News web page classification usingURLl content and structure attributes », in Next Generation Computing Technologies (NGCT), 2016 2nd International Conference on, 2016, p. 317–322.
[24] A. Alarabi et K. N. Mishra, « Artificial Neural Network Based Technique Compare with" GA" for Web Page Classification », in International Conference on Networked Digital Technologies, 2010, p. 699–705.
[25] A. Selamat et S. Omatu, « Web page feature selection and classification using neural networks », Information Sciences, vol. 158, p. 69–88, 2004.
[26] Y. Li, Y. Cao, Q. Zhu, et Z. Zhu, « A novel framework for web page classification using two-stage neural network », in International Conference on Advanced Data Mining and Applications, 2005, p. 499–506.
[27] G. Khade, S. Kumar, et S. Bhattacharya, « Classification of web pages on attractiveness: A supervised learning approach », in Intelligent Human Computer Interaction (IHCI), 2012 4th International Conference on, 2012, p. 1–5.
[28] W. Petprasit et S. Jaiyen, « E-commerce web page classification based on automatic content extraction », in Computer Science and Software Engineering (JCSSE), 2015 12th International Joint Conference on, 2015, p. 74–77.
[29] F. Feng, Q. Zhou, Z. Shen, X. Yang, L. Han, et J. Wang, « The application of a novel neural network in the detection of phishing websites », Journal of Ambient Intelligence and Humanized Computing, avr. 2018.

[30] G. Jain, M. Sharma, et B. Agarwal, « Optimizing semantic LSTM for spam detection », International Journal of Information Technology, vol. 11, no 2, p. 239-250, juin 2019.
[31] H. Yan, X. Zhang, J. Xie, et C. Hu, « Detecting Malicious URLs Using a Deep Learning Approach Based on Stacked Denoising Autoencoder », in Trusted Computing and Information Security, vol. 960, H. Zhang, B. Zhao, et F. Yan, Éd. Singapore: Springer Singapore, 2019, p. 372-388.
[32] D. López-Sánchez, A. G. Arrieta, et J. M. Corchado, « Visual content-based web page categorization with deep transfer learning and metric learning », Neurocomputing, vol. 338, p. 418-431, avr. 2019.
[33] D. López-Sánchez, A. G. Arrieta, et J. M. Corchado, « Deep neural networks and transfer learning applied to multimedia web mining », in Distributed Computing and Artificial Intelligence, 14th International Conference, vol. 620, S. Omatu, S. Rodríguez, G. Villarrubia, P. Faria, P. Sitek, et J. Prieto, Éd. Cham: Springer International Publishing, 2018, p. 124-131.
[34] S. R. Safavian et D. Landgrebe, « A survey of decision tree classifier methodology », IEEE transactions on systems, man, and cybernetics, vol. 21, no 3, p. 660–674, 1991.
[35] J. A. Mangai, D. S. Kothari, et V. S. Kumar, « A supervised discretization algorithm for web page classification », in Innovations in Information Technology (IIT), 2012 International Conference on, 2012, p. 226–231.
[36] H. Jelodar, S. J. Mirabedini, et A. Harounabadi, « Evaluation and Analysis of Popular Decision Tree Algorithms for Annoying Advertisement Websites Classification », in Communication Systems and Network Technologies (CSNT), 2015 Fifth International Conference on, 2015, p. 1025–1029.
[37] X.-Y. Lu, M.-S. Chen, J.-L. Wu, P.-C. Chang, et M.-H. Chen, « A novel ensemble decision tree based on under-sampling and clonal selection for web spam detection », Pattern Analysis and Applications, p. 1–14.
[38] M. Andoohgin Shahri, M. D. Jazi, G. Borchardt, et M. Dadkhah, « Detecting Hijacked Journals by Using Classification Algorithms », Science and Engineering Ethics, avr. 2017.
[39] M. Dadkhah, T. Sutikno, M. Davarpanah Jazi, et D. Stiawan, « An Introduction to Journal Phishings and Their Detection Approach », TELKOMNIKA (Telecommunication Computing Electronics and Control), vol. 13, no 2, p. 373, juin 2015.
[40] M. Dadkhah et G. Borchardt, « Hijacked Journals: An Emerging Challenge for Scholarly Publishing », Aesthetic Surgery Journal, vol. 36, no 6, p. 739-741, juin 2016.
[41] V. Estruch, C. Ferri, J. Hernández-Orallo, et M. J. Ramírez-Quintana, « Web Categorisation Using Distance-Based Decision Trees », Electronic Notes in Theoretical Computer Science, vol. 157, no 2, p. 35-40, mai 2006.
[42] J.-S. R. Jang, C.-T. Sun, et E. Mizutani, Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence. Upper Saddle River, NJ: Prentice Hall, 1997.
[43] C.-F. Tsai, Y.-F. Hsu, C.-Y. Lin, et W.-Y. Lin, « Intrusion detection by machine learning: A review », Expert Systems with Applications, vol. 36, no 10, p. 11994-12000, déc. 2009.
[44] A. D. Patel et Y. K. Sharma, « Web Page Classification on News Feeds Using Hybrid Technique for Extraction », in Information and Communication Technology for Intelligent Systems, vol. 107, S. C. Satapathy et A. Joshi, Éd. Singapore: Springer Singapore, 2019, p. 399-405.
[45] H. Zuhair et A. Selamat, « Phishing Hybrid Feature-Based Classifier by Using Recursive Features Subset Selection and Machine Learning Algorithms », in Recent Trends in Data Science and Soft Computing, vol. 843, F. Saeed, N. Gazem, F. Mohammed, et A. Busalim, Éd. Cham: Springer International Publishing, 2019, p. 267-277.
[46] A. Selamat et C. C. Ng, « Arabic script web page language identifications using decision tree neural networks », Pattern Recognition, vol. 44, no 1, p. 133-144, janv. 2011.
[47] G. Salton, A. Wong, et C. S. Yang, « A vector space model for automatic indexing », Communications of the ACM, vol. 18, no 11, p. 613-620, nov. 1975.
[48] S. Adam et B. Horst, Graph-theoretic techniques for web content mining, vol. 62. World Scientific, 2005.
[49] J. Kittler, M. Hater, et R. P. Duin, « Combining classifiers », in Proceedings of 13th international conference on pattern recognition, 1996, vol. 2, p. 897–901.
[50] B. L. J. Chuan, M. M. Singh, et A. R. M. Shariff, « APTGuard: Advanced Persistent Threat (APT) Detections and Predictions using Android Smartphone », in Computational Science and Technology, Springer, 2019, p. 545–555.
[51] X.-Y. Lu, M.-S. Chen, J.-L. Wu, P.-C. Chang, et M.-H. Chen, « A novel ensemble decision tree based on under-sampling and clonal selection for web spam detection », Pattern Analysis and Applications, p. 1–14, 2018.
[52] A. Makkar et S. Goel, « Spammer Classification Using Ensemble Methods over Content-Based Features », in Proceedings of Sixth International Conference on Soft Computing for Problem Solving, 2017, p. 1–9.
[53] F. Elsalmy, R. Ismail, et W. AbdelMoez, « Enhancing Web Page Classification Models », in Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2016, vol. 533, A. E. Hassanien, K. Shaalan, T. Gaber, A. T. Azar, et M. F. Tolba, Éd. Cham: Springer International Publishing, 2017, p. 742-750.
[54] L. Safae, B. E. Habib, et T. Abderrahim, « A Review of Machine Learning Algorithms for Web Page Classification », in 2018 IEEE 5th International Congress on Information Science and Technology (CiSt), 2018, p. 220-226.
[55] D. Boley et al., « Document categorization and query generation on the world wide web using webace », Artificial Intelligence Review, vol. 13, no 5-6, p. 365–391, 1999.
[56] M. Craven, A. McCallum, D. PiPasquo, T. Mitchell, et D. Freitag, « Learning to extract symbolic knowledge from the World Wide Web », Carnegie-mellon univ pittsburgh pa school of computer Science, 1998.

S.LASSRI received a software engineer degree from the National Institute of Statistics and Applied Economics (INSEA-Rabat) in 2014. She is a P.h.D student at Laboratory of Analysis Modeling and Simulation, Department of Mathematics and Computer Science, Faculty of Science Ben M'sik, Hassan II University Casablanca, Morocco



H.BENLAHMAR received a B.S. from Dhar lMahraz University - Fes in 1998, and an M.S. from the ENSIAS-Rabat in 2003. He received his Ph.D. in Computer Science from the ENSIAS-Rabat in 2007.
He is an Associate Professor in the Department of Mathematics and Computer Science at Faculty of Science Ben M'Sik Hassan II University, where he has been since 2008. Since 2016, he is a coordinator of the Master Data Science & Big Data. From 2014 to 2018 he served as Team Leader: Semantic Web and Knowledge Extraction at the TIM laboratory.

A.TRAGHA, Received a B.S. degree in applied mathematics from Mohammed V University, Morocco, 1983, and Doctorate of High Graduate Studies degree in Theories of Computer Sciences from Mohammed V University, Morocco, 1988 and Doctorate of state degree (or P. .D) July 2006 in Computer sciences from Hassan II University, Morocco. He is a Research Professor in the Department of mathematics and computer sciences of the Science faculty Ben M'sik, Hassan II University, Morocco since 1985. He was the coordinator of the Master from 2007 to 2015 and the Director of Modeling and Information Technology aoboratory from 2012 to 2016.
Research Area: Automatic Language Processing, Cryptography, Knowledge Engineering.