A Comparative Study of Parametric Versus Non-Parametric Text Classification Algorithms

Mihaela Chistol
Stefan cel Mare University of Suceava
Suceava, Romania
mihaela3milea@gmail.com

***Abstract*—Evolution of modern technologies allowed to store the text in various digital formats such as e-mails, e-documents, libraries, etc. The amount of text data that is produced daily is increasing dramatically. Discovering useful patterns in text that can be represented in unstructured, semi-structured or structured format is a difficult task that requires a good understanding of machine learning algorithms. Finding a suitable algorithm for text mining tasks such as classification, clustering or natural language processing is a demanding situation that tests researchers' abilities. This paper provides an overview of the text mining process also, presents a comparison of the performance and limitations of two predictive models generated using the parametric Naïve Bayes algorithm and non-parametric Deep Learning neural network. RapidMiner data science software platform has been used for models' implementations and e-mail classification.**

Keywords—text classification; text mining; machine learning; Naïve Bayes; neural network; performance evaluation.

I. INTRODUCTION

In the modern world, text is the most common way of sharing information. There is an increasing trend in the use of computers for storing information. As a result, 90% of the world's data are stored as unstructured documents. Therefore, proper classification and knowledge discovery, from huge amount of textual data, is an important area for research. Patterns discovery in text stream can be achieved through the combined use of Machine Learning and text mining techniques. Choosing the methods is a big challenge for the engineers because the application efficiency depends on the applied techniques.

Text mining is a modern technique for extracting knowledge from unstructured text through specific methods for patterns discovery. The name of this technique is an homage to data mining because it can be interpreted as a process of data mining that extracts text data. The main applications of text mining are Information Extraction (IE), Information Retrieval (IR) and Natural Language Processing (NLP). Companies take full advantage of this powerful technique to reduce repetitive tasks or to see if the customer review is positive or negative.

However, besides the obvious advantages, an inefficient algorithm can cause unpleasant situations, for example, an important email may not reach the recipient because it has been interpreted as a spam message. For this reason, the high accuracy of the models becomes an essential requirement for text mining tasks, and thus one of the best ways to improve it is to use an efficient algorithm.

Text classification techniques are evolving, their variety is constantly increasing, and with it grows the dilemma of choosing the right method for a task. To solve this dilemma, "autopsies" of the algorithms are performed by comparing: their type, suitability, representation schemes, the impact of feature reduction on the global performance accuracy, strengths, weaknesses, even their evolution over time. However, algorithms comparison is time-consuming and in the IT industry time is limited and well distributed. Often it is easier to choose a technique based on conclusions issued by other researchers. "Though there is voluminous literature stating the capabilities of different types of text classification techniques, the spread of these techniques in advanced fields like Artificial Intelligence (AI)/Machine Learning (ML) is seldom reported. Further, reviewing text classification approaches from an algorithmic point of view will benefit both the industry and academia equally" [1]. Thus, the focus of this study is to discover the best text classification techniques from a practical point of view, by making a comparison between different algorithms.

This work presents a comparative study made between two classifiers: a parametric one based on Naïve Bayes theorem and a non-parametric one based on Deep learning. For this purpose, two processes of knowledge discovery from text data were designed in RapidMiner Studio and for each of these, we used as training data the public set of SMS Spam Collection to generate two classification models that can predict the message's type (spam or ham, ham is an e-mail that is not spam, in other words it is "good mail").

The paper is organized as follows: Section 2 introduces the concept of knowledge discovery in data, Section 3 refers

This work was partially supported from the project "Integrated Center for research, development and innovation in Advanced Materials, Nanotechnologies, and Distributed Systems for fabrication and control", Contract No. 671/09.04.2015, Sectoral Operational Program for Increase of the Economic Competitiveness co-funded from the European Regional Development Fund.

978-1-7281-6870-8/20/$31.00 ©2020 IEEE

specific text preprocessing methods, Section 4 presents text classification models and Section 5 make a comparative performance evaluation. Finally, Section 6 summarize some conclusions and future works.

II. KNOWLEDGE DISCOVERY IN (TEXT) DATA

The knowledge discovery in textual data is not a new concept. In 1996 this was described as "nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data" [2]. It is a complex, iterative and interactive process having a strong interdisciplinary character.

Generally, the process consists in the following steps: business and data understanding, data collection, data preprocessing, data mining and finally model evaluation and interpretation. Text mining is a new approach of knowledge discovery in data in which, the data mining stage refers to unstructured data.

Fig. 1. presents a high-level overview of this process in which the main stages are: Feature Extraction, Dimensionality Reduction, and Classification.

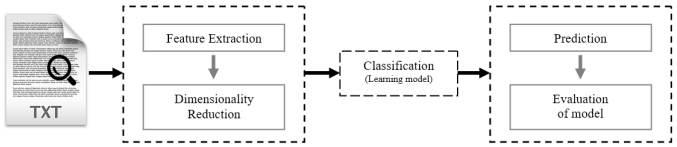

Fig. 1. An overview of text classification process.

Feature Extraction is the first step in which the unstructured text must be pre-processed and transformed into a structured feature space to be able to apply mathematical modeling classifiers. The well-known methods of feature extractions are Word2Vec [3], Term Frequency (TF) [4], Term Frequency-Inverse Document Frequency (TF-IDF), Global Vectors for Word Representation (GloVe) [5]. In the next section, we emphasize the importance of text pre-processing techniques such as tokenization, capitalization, noise removal, stemming, etc. and how they affect the correctness of the classification.

Dimensionality Reduction is the second stage in which researchers apply inexpensive algorithms to reduce time and memory. The well-known methods of dimensionality reduction are Linear Discriminant Analysis (LDA) [6], Principal Component Analysis (PCA) [7], t-distributed stochastic neighbor embedding (t-SNE), and non-negative matrix factorization (NMF).

The Classification stage is the most important and it requires a good understanding of each algorithm because the efficiency of the model depends on the chosen method. There is an ample range of text classification algorithms that can be grouped into distinct categories based on the learning procedure used. "Usually, a classification technique could be divided into statistical and machine learning (ML) approaches. Statistical techniques purely satisfy the proclaimed hypotheses manually, while ML techniques were specially invented for automation" [8]. In section IV we will present in detail the techniques used in text mining and we will identify the strengths and limitations of Naïve Bayes and Deep Learning algorithms by implementing two predictive models. For the research communities, it is very important to find a generalized solution for a type of problem. Thus, in this paper, a detailed comparison of the performances of the realized models is made to determine which machine learning algorithm is more suitable for the message's classification.

III. TEXT PRE-PROCESSING

Text mining is a technique that processes texts that are represented in a semi-structured, structured or natural language format. Data stored in such a format may contain redundant information, incomplete or missing values which will significantly affect the accuracy of the generated model. As the accuracy of the results depends on the correctness of the data used, pre-processing becomes one of the most important tasks that need to be done at the beginning.

For this study the public SMS Spam [9] data collection was used. This collection contains 450 spam messages collected from the Grumbletext Web site and 3375 ham messages randomly chosen from the NUS SMS Corpus. To clean this dataset new process was created in RapidMiner Studio Fig.2. and the following pre-processing techniques were applied:

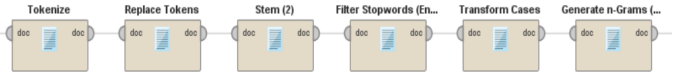

Fig. 2. Text pre-processing

- *Tokenization* is a technique that breaks a string of characters into phrases, words, or other significant elements. In Fig. 3. is shown the outcome of applying the Tokenize operator to a message.

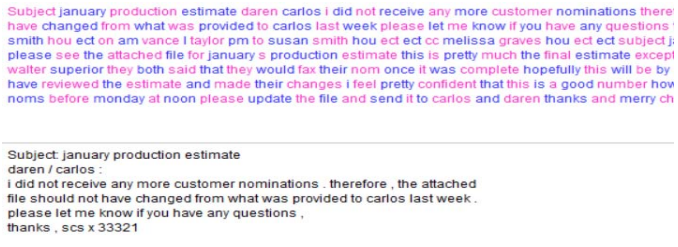

Fig. 3. The result of applying the Text Tokenize operator to a message.

- *Stemming* is one of the most important techniques. This method reduces related forms of a word to a common root i.e "studying" to the base "study". In this study, the stemming method was applied for English since all text data are in this language.

- *Filter Stopwords* is a technique that removes from the text words that do not bring a significant amount of information, such as the words “a”, “the”, “after” etc.
- *Capitalization* is used to transform text into lowercase. The original text may contain various forms of capitalization of the same word. Thus, it can cause difficulties in processing the text, which can be avoided by converting the text to uppercase or lowercase. This method can generate and inconvenience when the meaning of a word is changed, as capitalization changes. For example, “US” (United States of America) may be confused with the pronoun “us”. For this reason, capitalization should be applied only when it brings benefits.
- *N-Gram* method is a sequence of n-words that occurs in order in a sample of text. Sometimes words carry a different meaning when they are grouped for example, the word strategy has different meanings in word associations “military strategy” and “economic strategy”. By applying this statistical technique, we cannot extract the context from the document, but we can discover information about the frequency of common groups of words in the text. The N-Grams operator with max length parameter equal to 2 was used to apply this method, and a sample result is shown in Fig. 4.

Fig. 4. N-Grams

IV. CLASSIFICATION TECHNIQUES

Text classification intent is to classify text data into a determined number of categories. This is a difficult goal because text data can be inconsistent and unstructured. Therefore the algorithm is the key to a well-defined model. In articles of this research area, are mentioned various methods applicable in text mining. The significant approaches were arranged as a tree diagram, based on the algorithms learning procedures. In Fig.1, the algorithms are divided into two categories “Statistical” and “Machine Learning” the last is divided according to the learning criteria into “Supervised Learning”, “Unsupervised Learning” and “Semi-Supervised Learning”.

“Statistical techniques are purely mathematical processes, and they act as the mathematical foundation for all other text classifiers. It works similar to a computer program, executing the given instructions without any ability of its own” [1]. These methods are inefficient for large data sets and, for this reason, will not be used to classify emails.

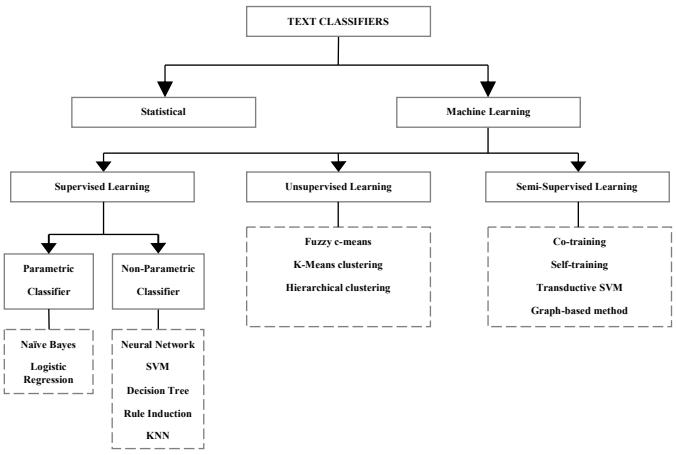

Fig. 5. Text Classifiers

Machine learning is an important part of text mining as it develops algorithms that automatically learn to make predictions about current data based on history. ML methods are divided into three distinct categories: supervised, semi-supervised and unsupervised. Usually, supervised learning techniques are employed for automatic text classification, these determine the result based on the knowledge acquired after processing the training data set. In recent years, the IT industry has paid more attention to text classification and the results have been observed rapidly, including in machine learning approaches. Two of these methods are described below.

A. Naïve Bayes

The Naïve Bayes classifier is founded on Bayes’s theorem, which was stated in the 18th century. It is a probabilistic algorithm with strong (naïve) independence assumptions between the features [11]. This technology began to be widely used for information retrieval and document classification since the 1950s. Nowadays, Naïve Bayes is considered a traditional method, recognized for its efficiency in classification problems.

In Fig. 6 is represented the process created for the implementation of the predictive model using the Naïve Bayes algorithm.

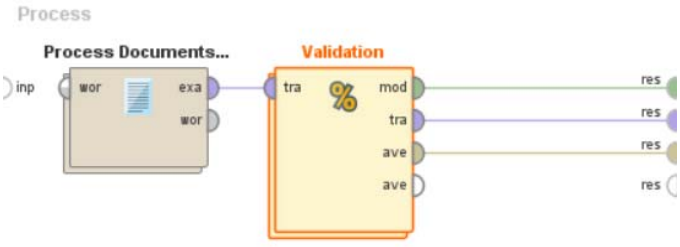

Fig. 6. The RapidMiner process of model implementation using the Naïve Bayes algorithm

Process Documents from Files operator is used for loading and preprocessing the dataset. The processed messages are provided as input to the Validation operator that randomly split the data into a training set and testing set. The Validation subprocess is presented in Fig. 7.

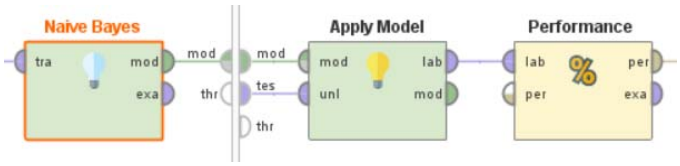

Fig. 7. Validation operator

In the training subprocess, the Naive Bayes operator is used for learning and building the model. "This operator uses Gaussian probability densities to model the Attribute data" [12]. The fundamental assumption of Naive Bayes is if the d fit into k categories, k ∈ $\{c_0, c_1, \ldots, c_n\}$, the classified class is c∈ C [13].

$$P(c|d) = \frac{P(c|d)P(c)}{P(d)} \quad (1)$$

In the testing subprocess, the model is applied to the test data, and then its performance is evaluated using the Performance operator.

B. Deep learning

State-of-the-art results in machine learning tasks, including natural language processing, have been achieved through the use of Deep Learning methods. The Deep Learning concept was injected in the machine learning community by Rina Dechter (professor of computer science in the Donald Bren School of Information and Computer Sciences at University of California), and to artificial neural networks by Igor Aizenberg (professor and Chair of the Department of Computer Science at Manhattan College) and colleagues in 2000, in the context of Boolean threshold neurons [14].

In text mining, this technology has three basic architectural representations: Deep Neural Networks (DNN), Recurrent Neural Network (RNN) and Gated Recurrent Unit (GRU). In this study, the DNN architecture was used to implement the predictive model.

"Deep neural networks are designed to learn by multi-connection of layers that every single layer only receives the connection from previous and provides connections only to the next layer in a hidden part" [13]. The standard Deep Neural Network model is presented in Fig. 8.

"Neural networks are very effective in cases where a hierarchical multi-label classification approach is required. This classification task is complex as each sample may belong to more than one class and predictions of one level are fed as inputs to the next level to make a final decision" [1].

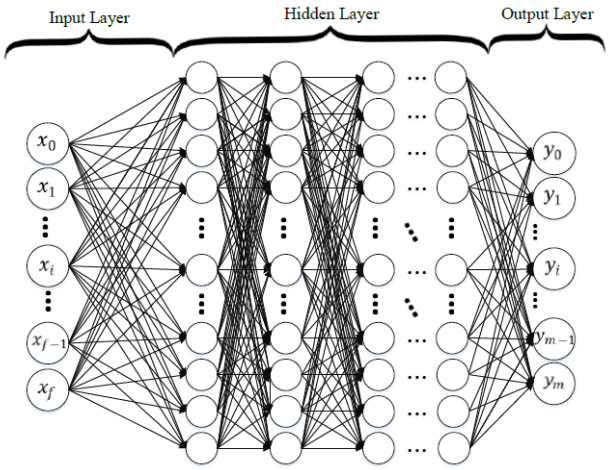

Fig. 8. Fully connected Deep Neural Network (DNN) [13]

In this work, the Deep Learning classification model is generated using an implementation of a multi-layer feed-forward neural network that passes the information through layers using the standard back-propagation algorithm. Given a set of inputs and a series of outputs the purpose of the algorithm is to determine a relationship between these values. In text classification, the input is a string that is processed by vectorizing primary data. Fig. 9. pictures the process created, in RapidMiner Studio, for the model implementation using this method.

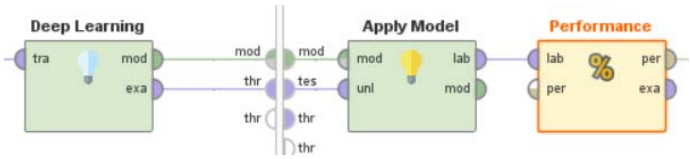

Fig. 9. The RapidMiner process of model implementation using the Deep Learning algorithm

Process Documents from Files operator is used for loading and preprocessing the dataset. The processed data are provided as input, via TF-IDF embedding, to the Validation operator that randomly split the data into a training set and test set. TF-IDF is a statistical method that shows the importance of a word in a text stream.

In the training subprocess, the Deep Learning operator is used for learning and building the model. The operator executes the DL algorithm using H2O 3.8.2.6. "H2O's Deep Learning is based on a multi-layer feedforward artificial neural network. A feedforward artificial neural network (ANN) model, also known as deep neural network (DNN) or multi-layer perceptron (MLP), is the most common type of Deep Neural Network and the only type that is supported natively in H2O-3" [15].

In the testing subprocess, by using the Apply Model operator, the model is applied to the test dataset. Statistical performance evaluation of the classification model is performed using the Performance operator.

V. COMPARATIVE PERFORMANCE EVALUATION

In the research community, algorithms comparison is a common practice that evolved significantly in the past years. There are many more techniques that can accomplish model evaluation but, in the text classification, the best practices remain the "key" to an accurate assessment. For example, even a simple act of choosing different training and testing datasets can lead to inconsistencies in model performance. Text classification metrics measure the model's ability to process a new dataset after it has been previously trained. The most used are accuracy, precision, classification error, micro-average, macro average, etc. The Performance (Classification) operator was used for statistical performance evaluation of models.

TABLE I. MODEL PERFORMANCE COMPARISON

Parameter	**Model performance comparison**		
	Detail	***Naïve Bayes***	***Deep learning***
accuracy	Accuracy is the metric that shows the percentage of dataset correctly classified.	97.81%	86.36%
classification error	Classification error shows the percentage of incorrect predictions.	2.19%	13.64%
absolute error	Absolute error is the average deviation of the prediction from the real value.	0.022+/-0.146	0.142+/-0.309
spearman rho	Spearman's rho is a metric that shows the relationship between label attribute and prediction attribute.	0.947	0.711
correlation	The metric that shows the correlation between the class attribute and the predictive variables.	0.947	0.711

After analyzing the performance metrics, it is clear, that the model implemented using the Naïve Bayes technique is more efficient, in the message classification tasks, than the model implemented using Deep learning. Even though Deep Learning is a forceful and efficient ML algorithm, it has some major disadvantages when it is applied in text mining. Since this technique does not work well on a small training set, a consistent dataset and high computing power were required. Thus, the model implementation time was 7 times greater than the time spent using the Naïve Bayes method.

TABLE II. CLASSIFICATION TECHNIQUES COMPARISON

Model	***Advantages***	***Limitations***
Naïve Bayes	✓ Very easy to implement ✓ Fast execution time ✓ Non sensitive to irrelevant features ✓ Works perfectly for text mining tasks	- Poor performance when attributes are strongly correlated
Deep learning	✓ Parallel processing capability ✓ Flexible with features design ✓ Handles complex input-output ✓ Architecture that can be easily adapt to different problems types	- Requires a substantial amount of training data - Requires a high computing power - Difficult model interpretation

VI. CONCLUSIONS

Over the past decade, text classification has become one of the most important tasks for the community. Machine Learning techniques allow researchers to perform this task in a better way.

However, engineers need to keep in mind that both parametric and non-parametric algorithms can have high performance only if they develop a solid perception of the dataset and feature extraction methods. Implementing an efficient classification model involves combining four types of techniques: feature extraction methods, dimensionality reduction methods, classification algorithms and in the final stage performance evaluation methods.

The steps listed above were used, in this study, to implement two predictive models aimed to detect spam messages. From the obtained results, it is obvious that the model implemented using the Naïve Bayes algorithm is more efficient than the model implemented using Deep Learning.

In the future work, we will analyze different text classification techniques employed in practice, their strengths and weaknesses, and also their applicability to unstructured, semi-structured, and structured text data.

ACKNOWLEDGMENT

This work was partially supported from the project "Integrated Center for research, development and innovation in Advanced Materials, Nanotechnologies, and Distributed Systems for fabrication and control", Contract No. 671/09.04.2015, Sectoral Operational Program for Increase of the Economic Competitiveness co-funded from the European Regional Development Fund.

REFERENCES

[1] M. Thangaraj, M. Sivakami, "Text Classification Techniques: A literature review", Interdisciplinary Journal of Information, Knowledge, and Management, India, vol. 13, pp. 117-135, May 2018. Available:https://doi.org/10.28945/4066

[2] U. Fayyad, G. Piatetsky-Shapiro and Smyth, "The KDD process for extracting useful knowledge from volumes of data", Communications of the ACM, pp. 27-34, 1996.

[3] Y. Goldberg, O. Levy, "Word2vec explained: Deriving mikolov et al.'s negative-sampling word-embedding method", 2014.

[4] G. Salton, C. Buckley, "Term-weighting approaches in automatic text retrieval", Inf. Process. Manag., vol. 24, pp. 513–523, 1988.

[5] J. Pennington, R. Socher, "Manning, C.D. Glove: Global Vectors forWord Representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)", Doha, Qatar, Doha, Qatar, vol. 14, pp. 1532–1543, October 2014.

[6] M. Sugiyama, "Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis", J. Mach. Learn. Res., vol. 8, pp. 1027-1061, 2007.

[7] H. Abdi, L.J. Williams, "Principal component analysis", Wiley Interdiscip Rev. Comput. Stat., vol. 2, pp. 433–459, 2010.

[8] M. Allahyari, S. A. Pouriyeh, M. Assefi, S. Safaei, E. D Trippe, J. B. Gutierrez, K. J. Kochut, "A brief survey of text mining: classification, clustering and extraction techniques", CoRR, 2017.

[9] D. a. G. C. Dua, "{UCI} Machine Learning Repository" University of California, Irvine, School of Information and Computer Sciences, 2017.

[10] T. Srivastava, "Difference between machine learning & statistical modeling.", 2015.

[11] S. Madeh Piryonesi, Tamer E. El-Diraby, "Role of Data Analytics in Infrastructure Asset Management: Overcoming Data Size and Quality Problems". Journal of Transportation Engineering, 2020.

[12] R. GmbH, "RapidMiner Documentation", RapidMiner, 2020. Available:https://docs.rapidminer.com/latest/studio/operators/validation/split_validation.html.

[13] Kamran Kowsari, Kiana Jafari Meimandi, Mojtaba Heidarysafa, Sanjana Mendu, Laura Barnes and Donald Brown, "Text Classification Algorithms: A Survey", Information, USA, vol. 10, pp. 150, April 2019.

[14] Igor Aizenberg, Naum N. Aizenberg, Joos P.L. Vandewalle, "Multi-Valued and Universal Binary Neurons: Theory, Learning and Applications", Springer Science & Business Media, 2000.

[15] H2O.ai, "Deep Learning (Neural Networks)", April 2020. Available:http://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/deep-learning.html

[16] A. Khan, B. Baharudin, L. H. Lee and K. Khan, "A Review of Machine Learning Algorithms for Text-Documents Classification," Journal Of Advances in Information Technology, vol. 1, 2010.