AnalyzeLab: a Tool to Help Machine Learning Developers Evaluating their Models

Wenhan Yang
Université de Toulouse
Toulouse, France
wenhan.yang53@gmail.com

Jing Zhai
Université de Toulouse
Toulouse, France
jing.zhai@ut-capitole.fr

ABSTRACT

We have developed a tool that aims at helping machine learning developers when defining the parameters of their models. This tool allows users to provide (pre-processed) training data under the form of sets of extracted features and labels, as well as test data sets on which the model should be trained/tested. The users get a visual view of the accuracy of the obtained model. The users can select the features as they want, to include in the model as well as the examples to consider during training and the examples to use in the evaluation. Several machine learning algorithms are available and can be chosen among (K-Nearest Neighbors, Naive Bays, SVM, and Decision Tree). This makes a very useful tool for developers which is interactive and visualised. This paper presents this tool.

CCS CONCEPTS

• **Information systems** → **Information retrieval**; *Content analysis and feature selection*; *Clustering and classification.*

KEYWORDS

Machine learning evaluation; information check-worthiness; Interactive evaluation; Development

1 INTRODUCTION

Machine learning is widely used for many tasks including in information retrieval (IR) tasks. When developing such models, an important part is the features used to train the model. Indeed, training is based on labelled examples that are represented by features or characteristics. Those examples are automatically analysed by the machine to elaborate a model that is trained to predict the appropriate decision (the label) both for training examples but also for unseen data. Another important part of machine learning is the algorithm used; and many algorithms have been developed in the literature, either to predict a value (e.g. regression) or to predict a class (e.g. Random Forest, Support Vector Machine) for any input.

In many cases developing and selecting features as well as choosing a ML algorithm also implies an evaluation process where the designers experiment features and algorithms. Evaluation measures how accurate the model is, either on the training data set, or on a test data set.

In this paper, we present a tool we developed in order to help designers and researchers when elaborating features. This tool allows a researcher to select the features, algorithms, and data sets (from files the user provides in a directory) s/he wants to evaluate. The application then trains the ML model and evaluates it. It shows the results to the user in a visual way that helps her/him to quickly analyze the impact of the model parameters.

The rest of the paper is organized as follows: Section 2 presents the use case we use to illustrate the system functionalities. Section 3 presents the different system functionalities throughout commented screen shots. Section 4 concludes this paper and mentions some possible future developments.

"Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0)."

Supported by Institut de Recherche en Informatique de Toulouse, CNRS UMR5505, France.

2 INFORMATION CHECK-WORTHINESS AS A USE CASE

Information check-worthiness task As an illustrative use case, we consider in this paper the information check-worthiness task introduced in CLEF 2018 evaluation forum [2]. The systems that answer this shared task predict whether a piece of information (a sentence from a political discourse) should be prioritized for truthfulness checking [6].

Information check-worthiness data sets The data sets we use in this illustrative use case were provided by (1) the task organizers with regard to the sentences to check and the ground truth [6] (2) Lespagnol *et al.* on demand regarding the features extracted from the above mentioned collection. Each short sentence, is represented by heterogeneous types of features: information nutritional label based on [3], linguistics, category hierarchy, and word-embedding based on Word2Vec model [5]; these features have been used in the CheckThat! shared task [1] and are described in [4]; they cannot be described in because of page limit.

Information Nutrition Label [3] is to help the online information consumer, proposing an Information Nutrition Label, resembling nutrition fact labels on food packages. Such a label describes, along a range of agreed-upon dimensions, the contents of the product (an information object) in order to help the consumer (reader) in deciding about the consumption of the object.

Both AnalyzeLab and Information Nutrition Label are creating for contributing less difficult to judge the trustworthiness of news found on the Web with the proliferation of online information sources. The difference between these tools is that Information Nutrition Label is more specifically applied by using natural language processing and AnalyzeLab is mostly concentrate on several machine learning algorithms. In the future we have the tendency to develop AnalyzeLab with NLP to apply further functions.

3 SYSTEM FUNCTIONALITIES AND ILLUSTRATIVE EXAMPLES

The tool allows users to choose (1) features, (2) training datasets, (3) test datasets and (4) the algorithms to be used for training/testing. At each run, the user can select one or several items of each category. Figure 1 presents the user interface.

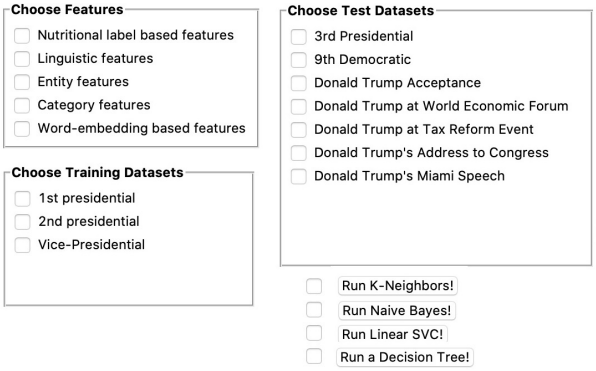
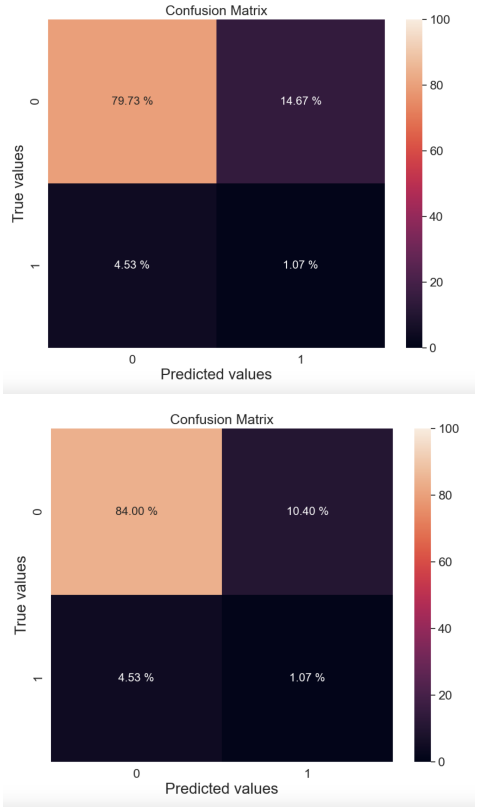


Figure 1: User Interface

This interface helps developers to figure out which features and algorithm suit better for the chosen dataset(s). Indeed, as an input, the interface provide the user with two types of results as follows:

- a colored confusion matrix which helps her/him to quickly have an overview of the accuracy of the run (see Figure 2);
- numerical results in terms of precision, recall, f1-score, quantity of the dataset, accuracy, macro avg and weighted avg for each class (See Figure 3)



(a) Using SVC algorithms (b) Using DT algorithm

Figure 2: Example of an heat map obtained

	precision	recall	f1-score	support
0	0.95	0.84	0.89	354
1	0.07	0.19	0.10	21
accuracy			0.81	375
macro avg	0.51	0.52	0.50	375
weighted avg	0.90	0.81	0.85	375
	precision	recall	f1-score	support

Figure 3: Detailed measures of the results corresponding to the heat map from 2 (a)

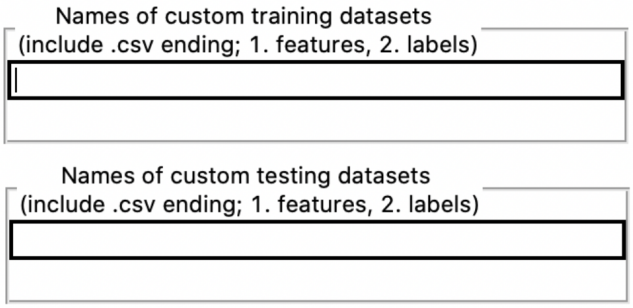


Figure 4: custome datasets input

We considered the case where the user chose the *linguistic features* as features among *Linguistic features*, *Entity features*, *Category features* and *Word-embedding based features* that are available. Also the user chose the 'Vice presidential' dataset as training dataset among the three that are available. The user chose 'Donald Trump's address to congress' as the test dataset among seven available for testing. Finally, the user selects the algorithms to use Linear SVC and Decision Tree.

We displayed confusion matrix and detailed measures of the results of the applied machine learning method. In the confusion matrix, "1" (resp. "0") for *Predicted values* corresponds to predicted check-worthy (resp. predicted not check-worthy). Similarly, "1" (resp. "0") for *True values* corresponds to labeled as check-worthy (resp. not labeled as check-worthy).

However, in the interface, the user can choose different features, training data sets and test data sets and different algorithms to compare the results.

We have implemented an option for custom dataset (See Figure 4). In order to apply user's datasets to train and test the model, user needs to provide a dataset that has same structure as ours. The first line is the full name of the user's training file and second line is the full name of the user's labeled data set. Depending on the ML problem, developers will use different features, and different data sets. In the current interface the boxes are not yet dynamically created but this is an extension that can be implemented in the future.

4 CONCLUSION AND FUTURE WORK

In this paper we have presented a tool that we believe could be very useful to developers when finalizing their ML models and features to include in a model. By interactively selecting the model parameter and visualizing the results, we make the development more simple. This tool could be expanded in various ways. First, we could add other evaluation measures than the confusion matrix for visual evaluation the user could choose among. Second, we could have a dynamic list of features automatically detected from costumed data sets and allow users to choose which features they want to use to train the model. Third, we could add some indicators that developers want for their model. For example, they could set an acceptable goal score for one or several indicators (e.g. Recall > 0.75) and the application could then go through all the possible combinations of features, algorithms and data sets in order to find the best model that matches the requirements if any.

ACKNOWLEDGMENTS

We would like to express our very great appreciation to Josiane Mothe for her valuable and constructive suggestions and supervision during the planning and development of this research work.

We also would like to thank Mickey Fraanje, Reynaldo Quintero, Manish Adhikari, Elijah Adeogun, Patrick Siekmeier, Amrutha Thalappan for their initial contribution to this tool.

REFERENCES

[1] Romain Agez, Clément Bosc, Cédric Lespagnol, Noémie Petitcol, and Josiane Mothe. 2018. IRIT at CheckThat! 2018. In *Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, Avignon, France.*

[2] Patrice Bellot, Chiraz Trabelsi, Josiane Mothe, Fionn Murtagh, Jian Yun Nie, Laure Soulier, Eric SanJuan, Linda Cappellato, and Nicola Ferro. 2018. Experimental IR Meets Multilinguality, Multimodality, and Interaction. In *Proceedings of the Ninth International Conference of the CLEF Association (CLEF 2018). Lecture Notes in Computer Science (LNCS)*, Vol. 11018. Springer.

[3] Norbert Fuhr, Anastasia Giachanou, Gregory Grefenstette, Iryna Gurevych, Andreas Hanselowski, Kalervo Jarvelin, Rosie Jones, YiquN Liu, Josiane Mothe, Wolfgang Nejdl, et al. 2018. An Information Nutritional Label for Online Documents. In *ACM SIGIR Forum*, Vol. 51. ACM, 46–66.

[4] Cédric Lespagnol, Josiane Mothe, and Md Zia Ullah. 2019. Information Nutritional Label and Word Embedding to Estimate Information Check-Worthiness. In *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*. ACM, New York, NY, USA, 941–944.

[5] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems 26*. Curran Associates, Inc., 3111–3119.

[6] Preslav Nakov and al. 2018. Overview of the CLEF-2018 CheckThat! Lab on Automatic Identification and Verification of Political Claims. In *Proceedings of the Ninth International Conference of the CLEF Association: Experimental IR Meets Multilinguality, Multimodality, and Interaction (Lecture Notes in Computer Science)*. Springer.