Health and Technology
https://doi.org/10.1007/s12553-020-00468-9

ORIGINAL PAPER

Greg, ML – Machine Learning for Healthcare at a Scale

Paola Lapadula[1] · Giansalvatore Mecca[2] · Donatello Santoro[2] · Luisa Solimando[1] · Enzo Veltri[3]

Received: 23 January 2020 / Accepted: 27 July 2020

Abstract
This paper introduces the Greg, ML platform, a machine-learning engine and toolset conceived to generate automatic diagnostic suggestions based on patient profiles. Greg, ML departs from many other experiences in machine learning for healthcare in the fact that it was designed to handle a large number of different diagnoses, in the order of the hundreds. We discuss the architecture that stands at the core of Greg, ML, designed to handle the complex challenges posed by this ambitious goal, and confirm its effectiveness with experimental results based on the working prototype we have developed. Finally, we discuss challenges and opportunities related to the use of this kind of tools in medicine, and some important lessons learned while developing the tool. In this respect, we underline that Greg, ML should be conceived primarily as a support for expert doctors in their diagnostic decisions, and can hardly replace humans in their judgment.

Keywords Machine learning for healthcare · Labeling tools · E-health · Digital patient records

1 Introduction

The push for the widespread adoption of digital records and digital reports in medicine [11, 17] is paving the ground for new applications that would not be conceivable a few years ago.

This paper presents one of these applications, called Greg, ML. Greg, ML [15] is a research project developed by Svelto! a spin-off of the data-management group at University of Basilicata. It is a machine-learning-based platform for generating automatic diagnostic suggestions based on patient profiles.

Machine learning is a well established branch of artificial intelligence. Supervised machine learning classfication algorithms [3], like, for example, logistic regression, neural networks and decision trees, are based on this approach: (i) developers identify a classification problem, i.e., the problem of tagging a collection of objects (in our case patient digital records, which we call *digital patient profiles*) with one or more labels (in our cases diagnoses); (ii) they collect a training dataset, i.e., a collection of objects for which labels are known in advance, and use this dataset to train one of the classical supervised ML algorithms (e.g., decision trees); (iii) the learning algorithm outputs a model, i.e., a piece of software that can be used to label new data objects; (iv) to validate the model, developers identify a test dataset, i.e., a collection of objects for which labels are known, but were not in the training dataset (that is, the test is performed on objects that were not used to train the algorithm); (v) f the quality of the labels over test data (e.g., precision and recall) are good enough, the model is deployed in a production environment, where it is used to predict, i.e., assign labels to new data objects that were not either in the training, nor in the test dataset, and for which labels are not known.

In essence, Greg, ML takes as input the digital profile of a patient, and suggests one or more diagnoses that, according to its internal models, fit the profile with a given probability. We assume that a doctor inspects these diagnostic suggestions, and takes informed actions about the patient.

We notice that the idea of using machine learning for the purpose of examining medical data is not new [13, 20, 21]. In fact, many efforts have been taken in this direction [7, 12, 23]. Greg, ML, however, is a distinguished effort in this landscape. In fact, all of the existing tools concentrate on rather specific learning tasks, for example identifying a single pathology – like heart disease [25, 30], or pneumonia [26], or cancer. For these very focused efforts, results of remarkable quality have been reported [31]. On the contrary, Greg, ML has the ambition of providing a broad-scope

✉ Enzo Veltri
enzo.veltri@gmail.com

1 Svelto! Big Data Cleaning and Analytics, Potenza, Italy

2 Università della Basilicata and Svelto! Big Data Cleaning and Analytics, Potenza, Italy

3 Università della Basilicata, Potenza, Italy

Published online: 22 August 2020

diagnostic-suggestion tool, capable, in perspective of providing suggestions about hundreds of pathologies. This, in turn, poses several complex challenges from the technology and procedural viewpoint, as discussed in the following sections.

The rest of the paper is organized as follows. We discuss Greg, ML's internal architecture in Section 2. Then, we introduce the infrastructure and the tools that support the generation of machine-learning models at a scale in Sections 3. A crucial element of the architecture is the labeler module, that supports the rapid development of training sets for the development of machine-learning models and we present it in Section 4. We discuss experimental results in Section 5. Related work is in Section 6. Finally, in Section 7 we conclude by listing some applications we envision for Greg, ML, and discuss a few crucial lessons learned with the tool.

2 Architecture of Greg, ML

The logical architecture and the main components of Greg, ML are depicted in Fig. 1. The core Greg, ML's architecture is built upon two different ML classifiers, constructed using the general approach described in Section 1:

- the main classifier is the Profile Classifiers and is used to label patient digital profiles with diagnoses; this is the model that generates outputs for doctors, i.e. it generates diagnostic suggestions;
- however, the system also uses a second, crucial ML classifier, i.e. the Report Classifier. This classifier takes a medical report for an instrumental exam – like, for example, X-Ray imaging – written in natural language, and labels it with what we call *pathology indicators* – e.g., pneumonia.

Patient profiles are entirely anonymous, i.e., Greg, ML does not store nor requires any personal data about patients. In the remainder of this section, we will describe the Greg, ML's architecture more in details.

2.1 Patient Profiles

The main building blocks of a patient profile are reports about clinical exams. Note that these fall in two categories:

- **result of lab exams** – like, for example, the level of glucose or sodium in blood;
- **textual reports from instrumental exams**, like RX, ultrasounds etc.

Notice that, while lab exam results come in numerical or categorical format, i.e., they are structured in nature and can be directly used within the profile, reports of instrumental exams are typically written in natural language and therefore essentially unstructured in nature. As a consequence, Greg, ML relies on a proper learning module to extract from textual reports what we call *pathology indicators*, i.e., structured labels indicating anomalies in the report that may suggest the presence of a pathology.

The Report Classifier is essentially a natural-language processing module. It takes the text of the report in natural language and identifies pathology indicators that are then integrated within the patient profile. This module is essential to the construction of the patient profile. In fact, reports of instrumental exams often carry crucial information for the purpose of identifying the correct diagnostic suggestions. At

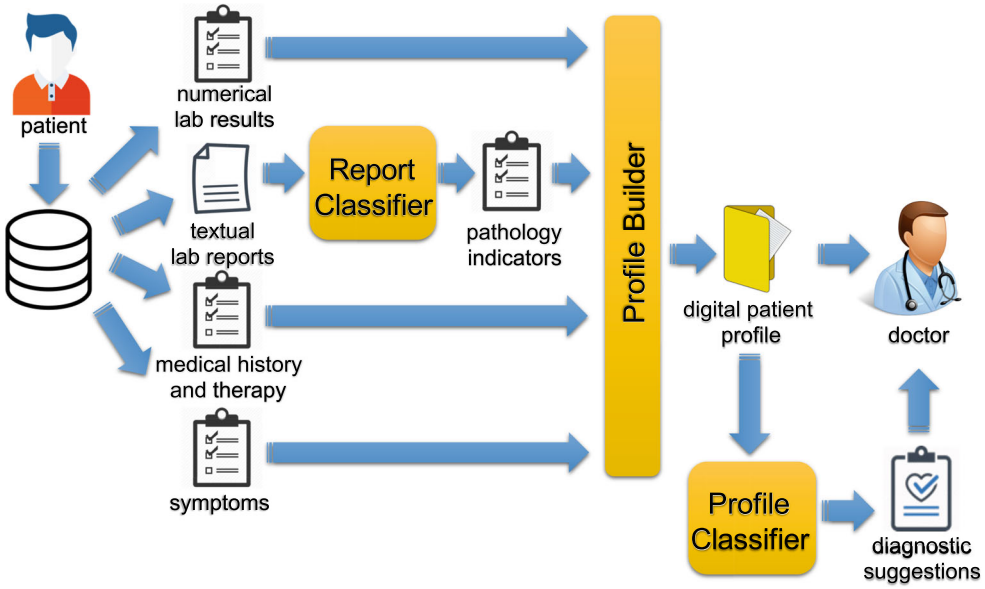

Fig. 1 Logical Architecture of Greg, ML. Patient data, with the pathology indicators generated using the Report Classifier, are feed to the Profile Builder that generates a patient profile. The patient profile then is feed to a Profile Classifiers that generate a set of diagnostic suggestions

the same time, their treatment is language-dependent, and learning is labor-intensive, since it requires to label large set of reports in order to train the classifier. We discuss in more detail the learning models adopted within the Report Classifier in Section 4.

Medical records represent the core of a patient's profile. However, the profile model has been designed to be modular since we soon learned that, whenever additional information is available, profiles can be properly enriched to make suggestions more sophisticated. More specifically, Greg, ML's profiles may optionally incorporate:

- **medical history of the patient**, i.e., past medical events and pathologies, especially chronic ones, or, alternatively, **current therapy**, i.e., drugs the patient is taking;
- **symptoms** declared by the patient at the time of admission.

There is in fact an inherent limitation about relying only on medical exams: these only capture pathologies that have not been treated yet, and therefore they seldom identify chronic pathologies. One prominent example is diabetes: diabetic patients assume insulin to balance their glucose levels. Therefore, lab exams do not show any irregularity in glucose levels. Whenever these are available, adding chronic pathologies or, alternatively, drug therapy to the patient profile allows the classifier to have a more accurate clinical picture.

Symptoms complement early-exam results and anamnesis in order to identify the directions to explore. We believe Greg, ML can be of particular help in ER, during the triage and first diagnostic phase; in particular, based on first evidences about the patient, it may help ER operators to identify a few pathologies that it is worth exploring, perhaps with the help of specialized colleagues. Greg, ML does not only suggest pathologies that are compatible with the patient profile, but also supports "what-if" reasoning. In particular, it is able to identify additional pathologies that might become relevant to the patient profile, provided that additional evidence – i.e., new reports about additional exams – is added to it. In these cases, Greg, ML reports these potentially relevant pathologies, and explicitly lists the set of new tests – e.g., blood exams – that need to be completed in order to explore the hypothesis.

All items listed above are collected by the Profile Builder module, which is responsible for composing a structured, digital profile of the patient. More in detail, a profile can be seen as a set of key-value pairs. Keys are of various types:(i) *personal data* like sex and age; (ii) *Tests from laboratory exams*, like 'Sodium' or 'Glucose' for blood tests; (iii) *Pathology indicators names* like 'Pleural Effusion' from RX tests; (iv) *Past Pathologies names* like diabetes; (v) *Drugs taken* like 'Insulin'; and (vi) *Symptoms names* like 'Fever' or 'Chest Pain'. The values depend on the types. Values for tests from laboratory exams are normalized values from the default ranges for each test. The values for the remaining keys are boolean values that state the presence or the absence of the related key. An example of a generated profile is shown in Table 1. Red lines indicate core fields, i.e., those coming from bio data and exam reports. Core fields are mandatory to create a learning profile. Greg, ML primarily relies on these to identify diagnostic suggestions. Yellow lines represent secondary fields, i.e., additional knowledge that can be added to the profile in order to improve the quality of suggestions.

The generated profile, then, can be fed to the Profile Classifier, which ultimately generates a set of diagnostic suggestions to be proposed to doctors based on decision trees. This proved to be a crucial design choice for the system, due to multiple reasons:

- first, doctors expect that medical suggestions provided by an automatic tool come with an explanation; they tend to trust these decisions as much as they "mimic" their way of thinking; on the contrary, they dislike "black-box suggestions", because they typically do not trust them; decision trees easily allow to provide explanations for a suggestion generated by the Profile Classifier, and therefore satisfy this requirement;
- most medical institutions rely on standardized clinical pathways or *integrated care pathways (ICPs)* to justify

Table 1 Example of a generated profile

Profile id: 12345	
Age	57
Sex	M
...	...
Glucose	3.48
Sodium	-1.73
Potassium	0.0
...	...
Pleural Effusion	false
Aortoscleroris	true
...	...
Diabetes	false
...	...
Insuline	false
...	...
Chest pain	false
Fever	true

Red lines indicate mandatory fields for a profile, like Age, Sex, available laboratory exams and all the pathology indicators that may be present or not. Yellow lines indicate past pathologies, drugs taken and symptoms. Yellow lines are not mandatory in the profile but if present they will increase the quality of the predictions

treatments and medical decisions; decision trees are the closest machine-learning model to ICPs; therefore Greg, ML may be used to quickly suggest which parts of a pathway need to be explored, and which ones can be excluded based on the available evidence;
- lastly, decision trees mimic the "medical-thinking" of the doctors that in the majority of the cases is based on $if - else$ conditions/statements.

There are a few important aspects to be noticed here:

- First, Greg, ML is trained to predict only a finite set of diagnoses. This means that it is intended primarily as a tool to gain positive evidence about pathologies that might be present, rather than as a tool to exclude pathologies that are not present. In other terms, the fact that Greg, ML does not suggest a specific diagnosis does not mean that can be excluded, since it might only be the case that Greg, ML has not be trained for that particular pathology. It can be seen that handling a large number of diagnoses is crucial, in this respect.
- Second, Greg, ML associated a degree of probability with each diagnostic suggestion, i.e., it ranks them with a confidence measure. This is important, since the tool may provide several different suggestions for a given profile, and not all of them are to be considered as equally relevant.

2.2 Deployment Architecture

The actual deployment architecture of Greg, ML's modules is depicted in Fig. 2.

It is easy to see that a tool like Greg, ML is as effective as seamless its integration with the everyday procedures of a medical institution is. To foster this kind of adoption, Greg, ML's classifiers have been wrapped under a REST API that can be easily integrated with any medical information system that is already deployed in medical units and wards. The Profile Builder is the only module that needs to be deployed with the hospital information system, in order to collect relevant data and build the profile to be sent to the REST API.

Ideally, with this kind of integration, accessing medical suggestions provided by Greg, ML should cost no more than clicking a button, in addition of the standard procedure for patient-data gathering and medical-record compilation.

Notice that Greg, ML is geared towards active learning. This means that, once a profile has been selected and suggestions have been generated by the Profile Classifier, doctors have a chance of providing feedbacks about each suggestion. This allows Greg, ML to collect further evidence and progressively refine its models.

We conclude by saying that in order to showcase its features and quickly provide a proof-of-concept of its effectiveness, we also developed a stand-alone Web app called the Greg, ML Playground.

3 The Greg, ML Ecosystem

As we have discussed in the previous sections, the effectiveness of a system like Greg, ML is strongly related to the number of pathologies which it can provide suggestions for. We therefore put quite a lot of effort in structuring the learning workflow in order to make it lean and easily reproducible. In this section we summarize a few key findings in this respect, that led us to the development of a number of additional tools, which compose the Greg, ML ecosystem.

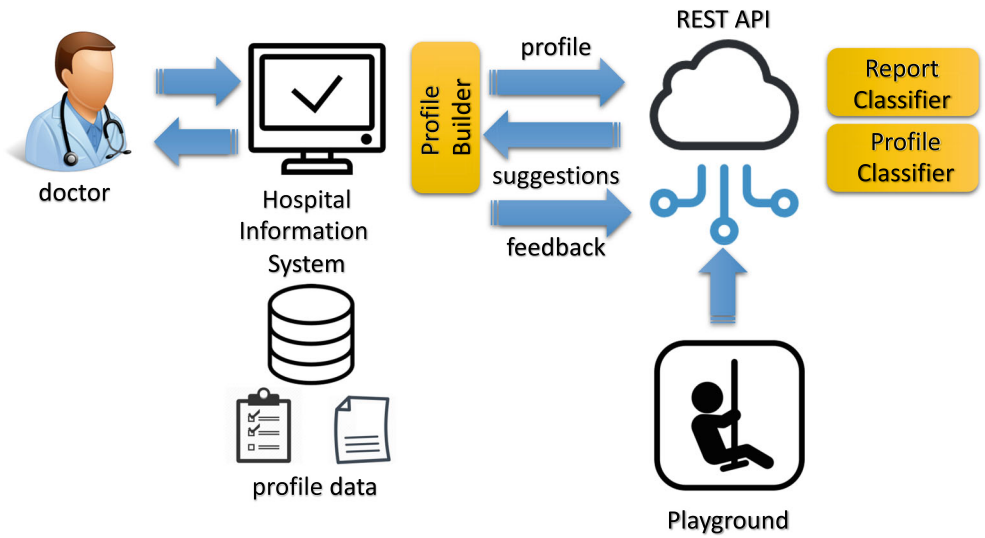

Fig. 2 Deployment Architecture of Greg, ML

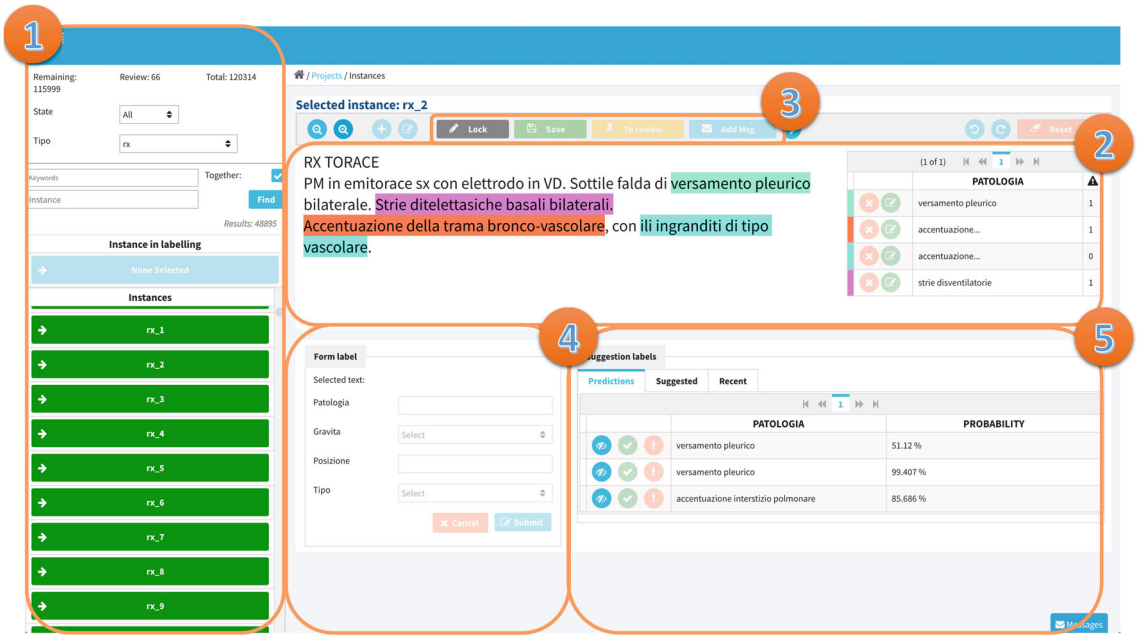

Fig. 3 DAIMO, the ML Labeling Tool. (1) Data Exploration of the samples; (2) Sample Labeling; (3) Collaborative and Clerical Review controls; (4) Thesaurus; (5) Automatic Label Suggestions

A first important observation we make is that a system like Greg, ML needs to refer to a standardized set of diagnoses. As it is common, we rely on the international classification of diseases, *ICD-10 (DRG)* [1]. This, however, poses a challenge when dealing with large and heterogeneous collections of medical records coming from disparate sources, which do not necessarily are associated with a DRG. In fact, standardizing the vocabulary of pathologies and pathology indicators is crucial in the early stages of data preparation. To this end we used a consolidated suite of data-cleaning tools [8–10].

A second important observation is that we need to handle large and complex amounts of data gathered from medical information systems, including admissions and patient medical history, medical records, multiple lab exams, and multiple reports. These data need to be explored, selected and prepared for the purpose of training the learning models. In order to streamline the data-preparation process, we decided to develop a tool to explore available data. The tool is called DWH and is essentially a data warehouse build on top of the transactional medical databases. This allowed us to adopt a structured approach to data exploration and data selection, that proved essential in the development of the tool.

However, the tool that proved to be the most crucial in the development of Greg, ML is DAIMO, our instance labeler, discussed in the next section.

[1] http://www.who.int/classifications/icd/icdonlineversions/en/

4 DAIMO: Annotations for Large-Scale ML

The main novelty of the platform we propose in this paper, and its main contribution, is the fact that it has been conceived to address automatic diagnostic suggestions on a large scale. To the best of our knowledge, there are no other proposals in the literature with these features. Other proposals have in fact a much more limited scope, and concentrate to a limited number, if not one, diagnosis.

DAIMO is a tool explicitly conceived to support the labeling phase of machine-learning projects.

DAIMO is a semi-automated tool for data labeling. It is not the first tool of this kind. For example, LabelMe [2] [27], is a free Web tool developed at MIT, exclusively tailored on labeling images. Both LabelBox [3] and Amazon Mechanical Turk [4] are commercial platforms that can be used to this end. However, DAIMO provides a set of unique features that set it apart from these tools.

To discuss this, we report a snapshot of the DAIMO interface in Fig. 3.

Sample Exploration To start, DAIMO provides a simple and effective interface to explore pre-defined collections of samples to label. As it can be seen in Fig. 3(1), users have at their disposal a rich set of controls to search, query and summarize the collection of samples. This is a crucial requirement for any training task that involves thousands of objects.

[2] http://labelme.csail.mit.edu/Release3.0/
[3] https://labelbox.com/
[4] https://www.mturk.com/

Sample Labeling Users can then pick a sample to label, explore its content and existing labels, and add more. Fig. 3(2) shows the process of labeling one textual lab report. Labels associated with the report are on the right. Each corresponds to a colored portion of the text. DAIMO allows users to add labels both to an entire instance, or to portions of it. This represents an important advantage with respect, for example, to LabelBox and Amazon Mechanical Turk, which to the best of our knowledge only support the first protocol.

Collaborative Labeling and Clerical Review DAIMO is aimed at collaborative labeling by a group of experts. It provides sophisticated control to lock, unlock, save and review labels Fig. 3(3). It supports both a single-step and clerical-review process. In single-step labeling, a user selects a sample, s/he adds labels, and these are saved into the database and considered as final. This protocol is acceptable only for toy applications. In mission-critical tasks a second review step is needed in order to reduce errors. Using this clerical-review protocol, user 1 adds labels and then sends the sample to review. A second user – different from the first one – will need to check the labels and either approve or reject them.

Shared Dynamic Thesaurus A crucial requirement, when working with a group of different experts, is the availability of a consensus dictionary of labels common to all users. Otherwise, a report stating that a patient has an enlarged hearth could be labeled by expert 1 with the "cardiomegaly" label, and by expert 2 with "enlarged cardiac shadow", thus making the two labels in conflict with each other.

We have soon learned that labels are not always known in advance. In large labeling applications they rather need to be constructed on the way. DAIMO allows users to define *label vocabularies*, in order to standardize the way in which labels are assigned to samples. Users usually search labels within the vocabulary Fig. 3(4), and add new ones only when the ones they need are not present. In our experience, this feature has proven crucial in order to get good-quality results with complex labeling tasks with many different labels

Label Suggestions We believe that even only the availability of an intuitive tool to support cooperative labeling-work, as discussed in the previous paragraphs, significantly increases productivity. However, the single most important innovation made by DAIMO is probably its suggestion-based labeling process. In essence, DAIMO incorporates an engine that is able to *learn* labeling strategies from examples. After some initial training, it does not only collects new labels from users, but actually suggests them, so that users need only to accept or refuse DAIMO's suggestions.

We have depicted the process of labeling instances with DAIMO in Fig. 4.

Users launch a new labeling project – in our example, labeling RX reports to identify aortosclerosis – by adding labels to a first bunch of samples. After a few dozen samples have been labeled, DAIMO enables suggestions. In essence, as shown in Fig. 3(5), when locking new instances to label, users find a set of suggested labels that they can accept or refuse. In this initial phase, DAIMO uses a trivial text-search strategy to provide suggestions. These are therefore of relatively low quality. To see this, assume users have introduced a "aortosclerosis" label. All samples containing this exact word will receive the label as a suggestion, regardless of its context. This will generate erroneous suggestions, like in "RX does not show signs of aortosclerosis". Still, in many cases the time needed to handle a new sample is reduced with respect to the one that would be needed to add every single label from scratch.

As soon as the number of labeled samples increases, DAIMO starts developing prediction models for them. After

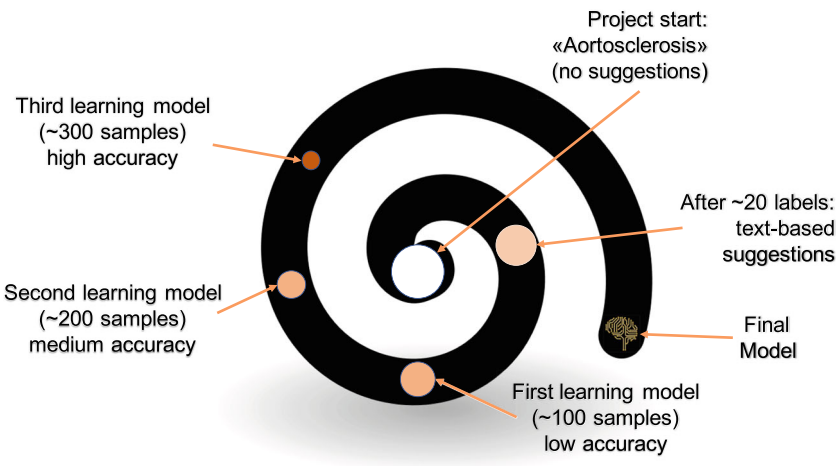

Fig. 4 Suggestion-based labeling with DAIMO. The spiral indicates the time to generate a ML model. The circles indicate the average time to label an instance. The lower is the diameter the lower is the average labeling time. The color of the circle indicates in a scale the accuracy of the model. White is no accuracy since there is no model, red is high accuracy

a label has been assigned 100 times, it develops a first model, by using a Naive Bayes classifier. This improves the quality of suggestions, that from now on are not based only on text occurrences, but rather on actual learning.

DAIMO supports *custom classifiers*, i.e. classifiers generated by data practitioners, and *generated classifiers*, i.e. the ones generated automatically by itself. For the generated classifiers all the tagged instances can be used to generate a dataset. This dataset is split automatically in training, cross-validation, and test set. DAIMO uses the cross-validation to auto-tune the hyperparameters using a grid search approach and use the test set to calculate the score of each generated model. Different strategies are used to generate the learning model and essentially depends on the size of the training data: I) when the dataset is small, the features can be automatically selected using the frequencies of the words, and then using the one-hot-encoding techniques the training data is fed to a Naive Bayes classifier; II) when the dataset is of medium size, the approach is like the previous one, but DAIMO also tries to use the Logistic Regression and verifies if there is an improvement against the Naive Bayes otherwise it will continue to use the Naive Bayes; III) with big dataset is possible to use Neural Networks and completely ignore the feature selection since it is done by the network, again checking the the results wrt. Naive Bayes and Logistic Regression.

As it can be seen in Fig. 4, the generation of predictive models is iterated as soon as the frequency of label increases. With higher number of samples, DAIMO uses an ensemble of models that include logistic regression and, for high numbers of samples, neural models. With the improvement of models, the accuracy of suggestions to users increases, so that after a while users simply need to accept the labels suggested by the system.

This approach really transforms the labeling process from the inside out, since after a while it is DAIMO and not the user to do most of the work. In fact, in our experience, working with DAIMO may lower text-labeling times up to one order of magnitude with respect to manual, unassisted labeling, as discussed in our experimental section. This is an essential building block of the Greg, ML vision, since it is the only way to increase the number of diagnoses handled by the system.

5 Experimental Results

Greg, ML currently supports 50 diagnoses. This section discusses several experimental evidences we got from our work with the system so far.

5.1 Dataset and Workable Profiles

To start, it is interesting to comment on the data we used for learning. We had access to an anonymized dataset composed of hospital records about from 2008 to 2017. Here are a few relevant figures:

- the dataset contains over 748.000 patient records, with a total of over 928.000 lab-exam results, and 112.000 instrumental exam reports;
- the number of admission reports complete with symptoms, medical history and therapy were very low, thus confirming that the core of the algorithm needs to focus on clinical exams;
- in this respect, the number of available records may appear as a large one; however, after data cleaning, the number of workable profiles – i.e., profiles for which both lab and instrumental exams were available – went down to approximately 142.000;
- while still promising, these profiles contained 1712 different diagnoses; we explored the frequency of these, to discover that the frequent ones, i.e., the ones with over 500 occurrences, were only 55.

It should be apparent why the current version of the system is limited to predicting 50 diagnoses. In essence, obtaining medical data is difficult. Obtaining high quality medical data is even more difficult. Ultimately, it is extremely hard to get medical data of high quality that can also be used for learning purposes.

5.2 Labeling Times and Costs

Most of the work we conducted on the data was related to labeling samples. This was especially important in order to develop models for textual reports and extract pathology indicators. We report a few indicators below.

We developed 35 different models for pathology indicators. In order to do this, our team of doctors labeled over 4200 samples. The availability of DAIMO was crucial in order to lower labeling times and streamline the process.

Figure 5 shows how labeling times with DAIMO decrease with time. It can be seen from the figure that times are initially relatively high. Then, they tend to decrease due to the availability of suggestions provided by DAIMO. The line of tendency shows that times may decrease up to one order of magnitude.

The overall average labeling time per sample for the 35 pathology indicators has been of approximately 2 minutes and 30 seconds. This time includes both initial specification of labels and then clerical review.

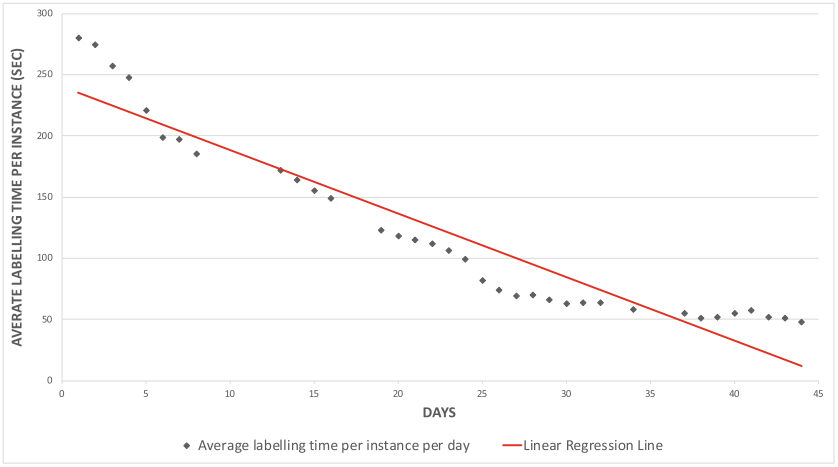

Fig. 5 Labeling times with DAIMO. Each point represents the average time in labeling an instance during the labeling phase. As the number of working days increases, and in essence the number of labeled instances increases, the quality of the suggested labels increased and so, the average time of labeling an instance decreased since the labeler will have only to accept the automatic suggestions. The red line indicates the trend of the average time

It is interesting to note that the overall time was 150 sec * 4200 samples, that is, less than 22 man days, i.e., approximately one man month. In terms of cost, this amounts to a few thousands euros. This is an important observation: on the one side, it supports our intuition that large-scale labeling of medical data has costs that may be considered economically feasible. On the other side, it again emphasizes the important of effective tools devoted to labeling.

5.3 Classification Quality

Let us discuss the quality of predictions, starting with the report classifier. We used the classical approach with training set, cross-validation set and test set. The dataset used is the one manually annotated by our team of doctors. We measured the F-Measure of the predictions of the report classifiers against the test set. Fig. 6 shows F-measures for the 35 pathology indicators. We consistently achieved F-measures over 90%.

Results of the profile classifier – the module responsible of ultimately providing suggestions to doctors – requires a specific discussion. One of the appealing features of our dataset is that it came with discharge letters. Therefore, we initially thought that it was possible to use as ground truth the set of diagnoses reported in the associated discharge letters.

Initial results, however, were significantly worse than we would have expected, especially for some diagnoses, like anaemia and urological infection. Our investigation of the data, however, showed us that Greg, ML performs better than the obtained results. In essence, in several cases Greg, ML suggested a more thorough set of diagnoses than the one indicated by the doctors in the discharge letter (see Table 2). As an example, anaemia is often associated with cirrhosis, and doctors often omit that from the discharge letter as show for Profile 1 in Table 2.

We therefore realized that also for patient profiles data needed labeling. We therefore asked our team of doctors to use DAIMO to review the set of diagnoses contained in

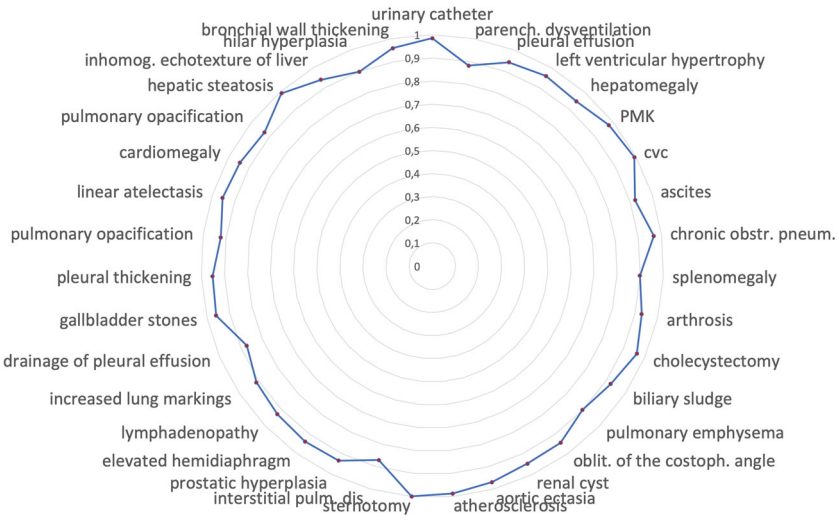

Fig. 6 Qualify of the report classificator (F-Measure)

Table 2 Example of predictions made by Greg, ML, labels that come from the Discharge Letters and labels that come from the manually annotated dataset made by our team of doctors. In bold the missing diagnoses from the Discharge Letters

Input Profile	Greg, ML Predictions	Discharge Letters	Manually Labelled by Doctors
Profile 1	Cirrhosis, **Anaemia**	Cirrhosis	Anaemia, Cirrhosis
Profile 2	**Anaemia**, Pneumonia	Pneumonia	Anaemia, Pneumonia
Profile 3	**Atherosclerosis**, Pleural Effusion, **Anaemia**	Pleural Effusion	Anaemia, Atherosclerosis, Pleural Effusion

discharge letters. In essence, our doctors made sure that all relevant diagnoses were appropriately mentioned, including those that the hospital doctors had omitted in the discharge letter (see the last column of Table 2). Using these improved dataset, Greg, ML achieves F-measure over to 95% for all 50 diagnoses.

In conclusion, these results show that diagnoses can be effectively learned from a noisy dataset where it is possible to have some missing labels. Also, they show that Greg, ML can effectively achieve high accuracy in its predictions. But this is possible only if enough data is available for the training purpose.

6 Related Work

In the last decade, a lot of proposals have been made to use Machine Learning in the field of medicine and healthcare [7, 12, 23]. Various approaches have been proposed: single diagnosis tools, general-purpose tools, and tools dedicated to medical imaging. In this section, we will review some new applications or proposals dedicated to the medical diagnoses.

Cardiovascular Medicine is one of the areas where numerous machine learning approaches have been proposed [14, 28], ranging from predictions of the survival of patients with HFpEF [29] to cardiovascular risk factor identification [19]. In some cases using deep learning techniques [18], like convolutional neural networks (CNN), AI systems were able to reach specialist's performance. As an example, a CNN with 34 layers was able to reach cardiologist performance in the arrhythmia detection [25].

A field considered only in recent years is the one concerning neurocognitive impairment in schizophrenia disorders. Recent studies have been proposed machine learning techniques as a tool to identify these specific cognitive deficits precisely of schizophrenia that characterize the disorder more markedly only through neurocognitive tests and as a tool to develop a predictive system based on the detection of the values of neurocognitive variables capable of diagnosing the presence of schizophrenia. Through the comparison between the same variables in healthy subjects and those with schizophrenia, carried out through the application of machine learning techniques, it is possible to perform, on an empirical and predictive basis, a more accurate diagnosis of schizophrenia on specific cognitive deficits [33].

Also, big companies like Google and IBM are involved with their artificial intelligence tools. For example, DeepMind health team (recently acquired by Google Health) uses medical data to provide better and faster health services, and, thanks to machine learning, it allows processing hundreds of thousands of medical information in a few minutes [22]. For example, DeepMind aims to speed up the planning of radiotherapy for patients. Currently, doctors need to manually create a map of the body parts to be treated and the healthy ones to avoid, in a process called "segmentation", which can take up to 4 hours (except for head-neck tumors) before radiotherapy. The algorithm developed by DeepMind, based on the interpretation of visual information in the body scans can speed up the process to 1 hour. Of course, since this tools are general-purpose, DeepMind is also involved in other different areas of healthcare predictions like the ability to predict acute kidney injury 48 hours before it happens [32], or in hospitalization prediction (the patient will stay long in the hospital, inpatient mortality and unexpected readmissions) [24].

IBM Watson, a cognitive system, is used in various fields of medicine as drug target identification and drug repurposing [4] or understand diabetes data [1], but the one that focuses most on is the fight against cancer with artificial intelligence. The system could be widely used from the match of patients to relevant clinical trials [1] or as support for doctors and oncologists in the ward to suggest more precise diagnoses and treatments for the treatment of tumors. Also, in this case, the algorithm is based on the study of images. Watson has limited capabilities because he is clearly not able to create "new knowledge" but can only be trained to connect the dots faster. In short, the doctor can be helped by the machine, but it is still indispensable.

Machine Learning could be used also as a back-office tool. In a recent work [16] a multilabel classifier was trained on the MIMIC-III database to learn ICD-9 diagnosis codes assignment using for each disease a model trained to learn disease-specific features.

In the area of the medical imaging, it is pushed a lot of effort in order to reach human performance in classification of the images. This is the case for example for systems that use deep-learning techniques as lung cancer prediction using

computed tomography [2], pneumonia prediction using x-rays [26] and diagnosis of retinal disease [6]. Another area where the use of machine learning is significant is cardiology. Cardiac imaging with magnetic resonance imaging (MRI) provides an accurate assessment of the functional status of the patient's heart. A recently published study [5] aimed to evaluate, in patients with pulmonary hypertension, whether right insufficiency and death can be predicted by using three-dimensional models and analyzing them automatically with supervised learning software.

What instead is proposed to do with Greg, ML is to make the machine learning system generic. We do not want to analyze single areas but try to make diagnostic suggestions on various pathologies. As seen above, the idea of using machine learning to examine medical data is not new. However, all the tools focus on rather specific learning tasks (identification of a single pathology). Greg, ML, on the other hand, has the distinctive characteristic of being a wide-ranging tool.

7 Conclusions: Opportunities and Lessons Learned

We believe that Greg, ML can be a valid and useful tool to assist doctors in the diagnostic process. Our experience sheds some light on the concrete possibility of developing a platform to learn diagnostic suggestions at scale, an effort that was not attempted before.

We believe that tools like Greg, ML may have numerous areas of applications. In addition to the ones mentioned above, like ER and first diagnosis, or ICP-based diagnosis, we imagine other possible use scenarios, as follows:

- We envision interesting opportunities related to the use of Greg, ML in the diagnosis of rare pathologies; these are especially difficult to capture by a learning algorithm, because, by definition, there are only a few training examples to use, and therefore a special treatment is required. Still, we believe that supporting doctors – especially younger ones, that might have less experience in diagnosing these pathologies – in this respect is an important field of application.
- Greg, ML may be used as a second-opinion tool, i.e., after the doctor has formulated her/his diagnosis, for the purpose of double checking that all possibilities have been considered.
- Finally, Greg, ML can be used to train young doctors that might have less experience in medical diagnosis.

While in our opinion all of these represent areas in which Greg, ML can be a valid support tool for the doctor, we would like to put them in context by discussing what we believe to be the most important lessons we have learned so far.

On the one side, the development of Greg, ML has taught us a basic and important lesson: in many cases, probably the majority, the basic workings of the diagnostic process employed by human doctors is indeed reproducible by an automatic algorithm.

In fact, it is well known that doctors tend to follow a decision process that looks for specific indicators within the patient profile – e.g., values of laboratory tests, or specific symptoms – and decides to consider or excludes pathologies based on them. As fuzzy as this process may be, as any other human-thinking process, to our surprise we learned that for a large number of pathologies this process provides a perfect opportunity for the employment of a machine learning algorithm, which, in turn, may achieve very good accuracy in mimicking the human decision process, with the additional advantage of scale – Greg, ML can be trained to learn very high numbers of diagnostic suggestions. In this respect, ironically quoting Gregory House, we might be tempted to state that "Humanity is overrated", indeed.

However, our experiences also led us to find that there are facets of the diagnostic process that are inherently related to intuition, experience, and human factors. These are, by nature, impossible to capture by an automatic algorithm. Therefore, our ultimate conclusion is that humanity is not overrated, and that Greg, ML can indeed provide useful support in the diagnostic process, but it cannot and should not be considered as a replacement of an expert human doctor.

Compliance with Ethical Standards

Conflict of interests The authors declare that they have no conflict of interest.

Funding Information Open access funding provided by Università degli Studi della Basilicata within the CRUI-CARE Agreement.

References

1. Ahmed MN, Toor AS, O'Neil K, Friedland D. Cognitive computing and the future of health care cognitive computing and the future of healthcare: the cognitive power of ibm watson has the

potential to transform global personalized medicine. IEEE pulse. 2017;8(3):4–9.
2. Ardila D, Kiraly AP, Bharadwaj S, Choi B, Reicher JJ, Peng L, Tse D, Etemadi M, Ye W, Corrado G, et al. End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography. Nature medicine. 2019;25(6):954.
3. Bishop CM. Pattern recognition and machine learning springer. 2006.
4. Chen Y, Argentinis JE, Weber G. Ibm watson: how cognitive computing can be applied to big data challenges in life sciences research. Clinical therapeutics. 2016;38(4):688–701.
5. Dawes TJ, de Marvao A, Shi W, Fletcher T, Watson GM, Wharton J, Rhodes CJ, Howard LS, Gibbs JSR, Rueckert D, et al. Machine learning of three-dimensional right ventricular motion enables outcome prediction in pulmonary hypertension: a cardiac mr imaging study. Radiology. 2017;283(2):381–390.
6. De Fauw J, Ledsam JR, Romera-Paredes B, Nikolov S, Tomasev N, Blackwell S, Askham H, Glorot X, O'Donoghue B, Visentin D, et al. Clinically applicable deep learning for diagnosis and referral in retinal disease. Nature medicine. 2018;24(9):1342.
7. Deo RC. Machine learning in medicine. Circulation. 2015;132(20):1920–1930.
8. Geerts F, Mecca G, Papotti P, Santoro D. Mapping and cleaning. In: Proceedings of the IEEE International Conference on Data Engineering - ICDE; 2014.
9. Geerts F, Mecca G, Papotti P, Santoro D. That's all folks! LLUNATIC goes open source. In: Proceedings of the International Conference on Very Large Databases - VLDB; 2014.
10. He J, Veltri E, Santoro D, Li G, Mecca G, Papotti P, Tang N. Interactive and deterministic data cleaning. In: Proceedings of the 2016 International Conference on Management of Data, SIGMOD Conference 2016; 2016. p. 893–907. https://doi.org/10.1145/2882903.2915242.
11. Heinis T, Ailamaki A. Data infrastructure for medical research. Foundations and Trends in Databases. 2017;8(3):131–238.
12. Holzinger A. Machine learning for health informatics. In: Machine learning for health informatics, pp. 1–24. Springer; 2016.
13. Kononenko I. Machine learning for medical diagnosis: history, state of the art and perspective. Artificial Intelligence in medicine. 2001;23(1):89–109.
14. Krittanawong C, Zhang H, Wang Z, Aydar M, Kitai T. Artificial intelligence in precision cardiovascular medicine. J Am Coll Cardiol. 2017;69(21):2657–2664.
15. Lapadula P, Mecca G, Santoro D, Solimando L, Veltri E. Humanity is overrated. or Not. Automatic diagnostic suggestions by Greg, ML (Extended Abstract). First international workshop on BIG data storage, processing and mining for personalized MEDicine, BIGPMED – ADBIS 2018 short papers and workshops; 2018. p. 305–313, https://doi.org/10.1007/978-3-030-00063-9_29.
16. Li Y, Chen W, Liu D, Zhang Z, Wu S, Liu C. Ifflc: an integrated framework of feature learning and classification for multiple diagnosis codes assignment. Ieee Access. 2019;7:36810–36818.
17. Miller RH, Sim I. Physicians' use of electronic medical records: barriers and solutions. Health affairs. 2004;23(2):116–126.
18. Miotto R, Wang F, Wang S, Jiang X, Dudley JT. Deep learning for healthcare: review, opportunities and challenges. Briefings in bioinformatics. 2017;19(6):1236–1246.
19. Miranda E, Irwansyah E, Amelga AY, Maribondang MM, Salim M. Detection of cardiovascular disease risk's level for adults using naive bayes classifier. Healthcare informatics research. 2016;22(3):196–205.
20. Mohammed O, Benlamri R. Developing a semantic web model for medical differential diagnosis recommendation. Journal of medical systems. 2014;38(10):79.
21. Peek N, Combi C, Marin R, Bellazzi R. Thirty years of artificial intelligence in medicine (aime) conferences: a review of research themes. Artificial Intelligence in Medicine. 2015;65(1):61–73.
22. Powles J, Hodson H. Google deepmind and healthcare in an age of algorithms. Health and technology. 2017;7(4):351–367.
23. Rajkomar A, Dean J, Kohane I. Machine learning in medicine. N Engl J Med. 2019;380(14):1347–1358.
24. Rajkomar A, Oren E, Chen K, Dai AM, Hajaj N, Hardt M, Liu PJ, Liu X, Marcus J, Sun M, et al. Scalable and accurate deep learning with electronic health records. NPJ Digital Medicine. 2018;1(1):18.
25. Rajpurkar P, Hannun AY, Haghpanahi M, Bourn C, Ng AY. Cardiologist-level arrhythmia detection with convolutional neural networks. 2017. arXiv:1707.01836.
26. Rajpurkar P, Irvin J, Zhu K, Yang B, Mehta H, Duan T, Ding D, Bagul A, Langlotz C, Shpanskaya K, et al. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. 2017. arXiv:1711.05225.
27. Russell BC, Torralba A, Murphy KP, Freeman WT. Labelme: A database and web-based tool for image annotation. Int. J. Comput. Vision. 2008;77(1-3):157–173. https://doi.org/10.1007/s11263-007-0090-8.
28. Seetharam K, Shrestha S, Sengupta PP. Artificial intelligence in cardiovascular medicine Curr Treat Options Cardio Med. 2019. https://doi.org/10.1007/s11936-019-0728-1.
29. Shah SJ, Katz DH, Selvaraj S, Burke MA, Yancy CW, Gheorghiade M, Bonow RO, Huang CC, Deo RC. Phenomapping for novel classification of heart failure with preserved ejection fraction. Circulation. 2015;131(3):269–279.
30. Soni J, Ansari U, Sharma D, Soni S. Predictive data mining for medical diagnosis: an overview of heart disease prediction. International Journal of Computer Applications. 2011;17(8):43–48.
31. Steadman I. IBM's Watson is better at diagnosing cancer than human doctors. 2013. http://www.wired.co.uk/article/ibm-watson-medical-doctor.
32. Tomašev N., Glorot X, Rae JW, Zielinski M, Askham H, Saraiva A, Mottram A, Meyer C, Ravuri S, Protsyuk I, et al. A clinically applicable approach to continuous prediction of future acute kidney injury. Nature. 2019;572(7767):116–119.
33. Vacca A, Longo R, Mencar C. Identification and evaluation of cognitive deficits in schizophrenia using machine learning Psychiatria Danubina. 2019.