IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 8 March 2023, accepted 17 October 2023, date of publication 26 October 2023, date of current version 7 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3327913

RESEARCH ARTICLE

A Cascade of Learners for Firemen' Emergency Events Classification

CAMILLO MARIA CARUSO[1], PAOLO SODA[1], (Member, IEEE), CARLO GIAMMICHELE[2], FRANCESCA ROTILIO[2], AND ROSA SICILIA[1]

[1]Research Unit of Computer Systems and Bioinformatics, Department of Engineering, Università Campus Bio-Medico di Roma, 00128 Rome, Italy
[2]IT Services of Italian National Fire Corps, 00184 Rome, Italy

Corresponding author: Camillo Maria Caruso (camillomaria.caruso@unicampus.it)

This work was supported in part by the University-Industry Educational Centre in Advanced Biomedical and Medical Informatics (CEBMI) (Educational, Audiovisual, and Culture Executive Agency of the European Union) under Grant 612462-EPP-1-2019-1-SK-EPPKA2-KA; and in part by the Piano Nazionale di Ripresa e Resilienza (PNRR) Ministero dell'Università e della Ricerca (MUR) Project under Grant PE0000013-FAIR.

ABSTRACT The main objective of firefighters is to optimise readiness in response to hazardous events and to minimise their collateral effects. In this context, few but growing research is investigating machine learning algorithms to support firefighters' work. Hence, this paper presents a decision support system to promptly identify *relevant* interventions, which are those events for which the national control room needs to alert the competent authorities because they could be dangerous for the community. The aim is to provide firefighters useful information for the management of such interventions and of the available resources. We define a set of new hand-crafted features specifically designed for the task, which catch both static and dynamic characteristics of the events. Furthermore, we design a learning approach based on a cascade of binary classifiers, which exploits the ability of most of the available classification algorithms to learn binary functions and it takes advantage of some characteristic of the dataset. The experiments were performed in leave-one-out on a real-world data set provided by the Italian National Fire Corps, analysing how the system works to distinguish among relevant and not-relevant interventions, both at the time of the call and during the events updates. The results show that machine learning-based decision support system significantly outperform the human operator.

INDEX TERMS Firefighting operations, supervised learning approaches.

I. INTRODUCTION

The National Fire Corps (Corpo Nazionale dei Vigili del Fuoco - CNVVF) ensures the safety of people, the integrity of property and environment as well as it provides technical interventions, including highly specialised content and appropriate instrumental resources. Within the institutional competencies of the National Fire Corps, the function of pre-eminent public interest is fire prevention. To this end, the study, the preparation and the experimentation of standards, measures, actions aimed at preventing the occurrence of a hazardous event or limiting its consequences play a fundamental role. Regardless of the nature of the event, the requirements for successful intervention by the firefighters include the immediacy of the service, the proper deployment of available resources and coordination with other law enforcement agencies.

For the CNVVF it is essential to identify the interventions for which the national control room must inform the competent authorities, such as the head of the Home Affairs Office, the Civil Protection Department, the National Police, etc.. These events are referred to as *relevant* and, for instance, they could have major consequences on the population, on some historical or artistic buildings or even on cities' public transportation or main pathways. Straightforwardly, in the following we will refer to the other events as *not-relevant*. In this respect, the current practice starts with the fire corps' operator receiving the emergency call for a given event: he/she identifies the hazard and assesses the situation, and then gives rise to the intervention needed to resolve

The associate editor coordinating the review of this manuscript and approving it for publication was Jenny Mahoney.

such an event. Next, he/she labels the event as relevant according to his/her experience and using a vademecum, i.e., a document listing several conditions that should be satisfied to make relevant the event and the associated intervention. Nevertheless, some events last several hours or days and, in these cases, *updates* are collected over time: such new pieces of information can lead the events to be assigned to the relevant class. Finally, the national control room manages and monitors those events received from the operators, and it alerts the competent authorities when needed. However, this approach leads to an excessive generation of relevant events from the operators, that, in turn, corresponds to a high false positive rate, assuming that the relevant class is the positive one. Indeed, according to the data available in this work and discussed hereinafter, this results in a sensitivity slightly larger than 10%.

To tackle this limitation, here we present an artificial intelligence (AI)-based decision support system (DSS) that automatically identifies the relevant interventions, an issue that has never been tackled in the literature to the best of our knowledge. Nevertheless, some researchers and practitioners have recently started investigating the use of AI to develop services able to support firefighters' activities in the different phases of the emergency, such as the preparation, the response, the recovery and the mitigation [1]. In our case, the identification of relevant events would help the control room to develop services for better management of the resources and to reduce the time required to identify an appropriate response, so that we fall within both the preparation and the response phases. In these two areas contributions in the literature are few but growing in number [2], [3], [4], [5], [6], [7], [8], which are summarized in Table 1 where we highlight the aims, the data, the learning models adopted and the performance achieved for each contribution. In [2] and [7] the authors forecast the time needed to reach the intervention area using data collected in a private dataset integrated with driving time estimates computed using distance information retrieved from an online routing engine (e.g., Google Maps), whereas [8] predicts both the victims' mortality and their need for transportation to health facilities, in order to better allocate their available resources. Except for these three works, all the others listed in Table 1 aim to predict the number of interventions in a given time slot that in most of the cases is one hour. To this end, they use data from private datasets that feed learning models belonging to different paradigms, such as neural networks, ensemble of classifiers and regression approaches. With respect to this overview of the literature, [8] could be regarded as the most similar work compared to our contribution. However the main aim of the authors is to predict the mortality of victims, which is substantially different considering our definition of relevant events, that take into account the different types of events that could harm public safety and order.

To sum up, our work presents three main contributions:

- we develop a DSS to automatically identify relevant interventions, to potentially optimise CNVVF preparation and response to such events executing the necessary actions;
- we define several features relying on both static and dynamic quantities computed from the fusion of data collected by the firefighters' operator at the time of the call with geographical information about the event locus as well as with the data on the resources used for the intervention on the field;
- we further power our DSS with Explainable AI (XAI) to determine the features that most influenced the predictions and provide further information to the operators that would potentially use this system.

The rest of the paper is organised as follows: section II illustrates the available materials and the pre-processing methods applied. Section III defines the proposed classification approaches and the computed features. Section IV provides details on the experimental setup used. Finally, sections V and VI present the results and derive some conclusions that can guide further studies.

II. MATERIALS AND DATA PREPROCESSING

Firefighters are not only required for fires, but they also have to deal with many types of events, such as non-conventional risks arising from criminal acts, preparation of national and territorial civil defence plans and intervention in case of landslides, floods or other public disasters.

The National Fire Corps provided two datasets: (i) the CNVVF dataset, which contains a random extraction of 2264 interventions carried out in the years 2016 and 2017; (ii) the SMS dataset, which contains 7554 messages sent for any purpose by the National Fire Corps control room to the local authorities.

This means that the latter repository contains both messages used for generic communication aims and messages used to inform about relevant interventions where, here, the term relevant is used according to the definition introduced in section I. Table 2 shows the 25 attributes available in the CNVVF dataset with the rate of missing values, which are imputed as presented in the supplementary material.[1]

In order to automatically identify relevant interventions we implement an algorithm that links messages of the SMS dataset to the interventions in the CNVVF dataset, which is discussed with more detail in the supplementary material as its presentation is not central with respect to the scope of this work. In summary, we first pre-filter each message in the SMS dataset by looking for candidate records in the CNVVF dataset that match according to different criteria. For instance, such criteria compare the intervention time slot in the CNVVF dataset with the time the emergency team initiated the intervention in the SMS dataset, or it matches the description keywords in the CNVVF dataset with the words used in the SMS messages. If the set of candidate records of a message is empty, we discard this message since

[1] As shown in Supplementary Table 1, the original dataset accounted for 34 attributes, but 9 are neglected since they are missing for almost all the samples or they contain redundant information.

TABLE 1. Summary of related work dealing with AI-based system to better manage the resources and to reduce the time required to identify an appropriate response. The symbol # denotes the number of interventions. MAE and RMSE stand for mean absolute error and root mean square error, respectively, while min and h are minutes and hours.

Article	Aim	Data	Learning Model	Performance
[2]	To predict firefighters' arrival time in real-time.	• SFFD call history dataset publicly available in Google Big Query. • Driving time estimates from the Google Maps Distance Matrix API.	• Linear Regression, optimised with stochastic gradient descent • Elastic Net Regularization • Decision Tree Regression • Random Forest	$MAE = 3.50\ min$ $RMSE = 5.00\ min$ for all the learning models
[3]	To predict the number of interventions using different time step configurations for the next one and three hours in the future.	List of interventions of firefighters in the Doubs (France) collected from 2012 to 2017.	LSTM model	• 1 hour prediction with a time step of 3h: $RMSE = 2.30\#/h$, $MAE = 1.69\#/h$ • 3 hours prediction with a time step of 24h: $RMSE = 4.22\#/3h$, $MAE = 3.19\#/3h$
[4]	To predict the number of interventions per hour.	List of interventions of firefighters for the period 2012-2017 in the Department of the Doubs, France.	Multilayer Perceptron	$MAE = 1.81\#/h$
[5]	To predict the number of firemen interventions in the next hour.	List of interventions from 2006-2018 provided by the department fire and rescue SDIS25 in Doubs-France.	• AdaBoost • Gradient Boosting • XGBoost	$MAE = 1.80\#/h$ $RMSE = 2.50\#/h$ for the XGBoost (best model)
[6]	To predict the number and types of firefighters' interventions by region.	List of interventions from 2006-2018 provided by the department fire and rescue SDIS25 in Doubs-France.	XGBoost	$MAE = 1.92\#/day$ $RMSE = 2.68\#/day$
[7]	To predict ambulances' response time to allow a dynamic selection of ambulance dispatch centers.	List of emergency medical services interventions from 2006-2020 provided by the department fire and rescue SDIS25 in Doubs-France	• Light Gradient Boosted Machine • XGBoost • Multilayer Perceptron • Least Absolute Shrinkage and Selection Operator	$MAE = 3.43\ min$ $RMSE = 5.54\ min$ for the XGBoost (best model)
[8]	To predict both the victims' mortality and their need for transportation to health facilities.	List of interventions from 2015-2020 provided by the department fire and rescue SDIS25 in Doubs-France.	• Light Gradient Boosted Machine • XGBoost • Convolutional Neural Network • Attentive interpretable tabular learning (TabNET)	• Victim's mortality: $ACC = 96.44\%$, $AUC = 96.04\%$ • Need for transportation to health facilities: $ACC = 73.62\%$, $AUC = 78.91\%$ for the XGBoost (best model).

TABLE 2. List of attributes available in the CNVVF dataset listed by category, showing also their type and rate of missing values. The *date timestamp* refers to a structure like DD/MM/YYYY, just as the *time timestamp* to HH:MM.

Category	Attribute	% of missing values	Type	Description
Temporal	intervention number	0.00	integer	Intervention identification number.
	intervention date	0.00	date timestamp	Date of intervention.
	call date	0.00	date timestamp	Date of call.
	call time	0.00	time timestamp	Time of call.
	first out time	0.00	time timestamp	Time of departure of the first emergency vehicle.
	service date	0.00	date timestamp	Date of service related to the operating team.
	assignment date	0.00	date timestamp	Date of assignment to the operating team.
	assignment time	0.00	time timestamp	Time of assignment to the operating team.
	exit time	10.79	time timestamp	Time of departure from the station of the operating team.
	departure date	0.00	date timestamp	Date of departure of the operating team from the event site.
	departure time	32.68	time timestamp	Time of departure of the operating team from the event site.
	return date	0.00	date timestamp	Date of return to the station of the operating team.
	return time	19.83	time timestamp	Time of return to the station of the operating team.
	closing date	0.00	date timestamp	Closing date of the event, the step following the return of all teams involved.
	closing time	0.00	time timestamp	Closing time of the event, the step following the return of all teams involved.
Geographical	road name	0.00	string	Name of the street of the event.
	municipality	0.00	string	Name of the municipality of the event.
	province abbreviation	0.00	string	Abbreviation of the name of the province.
Resources	description	0.00	string	Description of the intervention, chosen from a list of possible values.
	typology detail	28.16	string	Details describing the nature of the intervention, manually entered.
	vehicle type	0.00	string	Type of emergency vehicle involved.
	relevant target	0.00	boolean	Label indicating the relevance of the target followed by the operator's notification to the operational centre.
	relevant intervention	0.00	boolean	Label indicating the relevance of the intervention followed by the operator's notification to the operational centre.

it refers to an intervention not included in the CNVVF dataset. Nevertheless, searching for candidate records may return a list of samples from the CNVVF dataset, but only one has to be selected. To address this issue, for each SMS message and for each of its candidate records, we compute a score based on the number and type of information that match between them. Straightforwardly, we assign the SMS message to the candidate record from the CNVVF dataset with the largest score, excluding those with intermediate scores, and the selected record in the CNVVF dataset is assigned to the *relevant* class. Furthermore, we discard the SMS messages if all its candidate interventions get a zero score. Finally, we assign to the *not-relevant* class all the interventions in CNVVF dataset not assigned to any SMS message.

As already mentioned in section I, there exist interventions in the CNVVF dataset originally labelled as relevant by the operator receiving the emergency call which have not been followed by an SMS message. Although they are assigned to the *not-relevant* class, we mark them as *interesting*, effectively introducing a third class. We deem that this could be helpful from a machine learning perspective because relevant and interesting interventions have similar characteristics that not only misled the human operator but could make overlap the samples in the feature space, thus making more difficult the training process. The introduction of this artificial and additional class should allow the model to separately learn the distinctive features of these interventions. These assumptions are further confirmed by the obtained results, as it will be shown in section V.

At the end of the process described in this section we got 708, 360 and 1196 interventions in the relevant, interesting and not-relevant classes, respectively. When treated as a binary test, as humans would do, this means that we have 708 and 1556 relevant and not-relevant events, respectively.

III. METHODS

The proposed learning approach leverages two different types of features, referred to as *static* and *dynamic*. The former do not change with the evolution of the event, even if the intervention lasts several hours or days, as described in section I. The latter are computed at each update of the intervention, capturing its temporal evolution.

While these descriptors are presented in details in section III-B, let us now introduce the following notation:

- m is the number of training samples;
- u is the number of static features;
- v is the number of dynamic features;
- $\mathbf{l}^{tr}$ is the matrix $m \times 3$ containing the ground truth labels of the training samples according to a 1-of-N coding;
- $\mathbf{X}^{s,tr}$ is a matrix $m \times u$ including the static features, as denoted by the apex s. Furthermore the apex tr stands for the training set and, straightforwardly, when it is replaced by te denotes the test sample, so that $\mathbf{X}^{s,te}$ is a $1 \times u$ row vector;
- n is the number of updates;
- $\mathbf{X}_i^{d,tr}$ is a matrix $m \times v$ including the dynamic features, as denoted by the apex d, with $i \in \{0, \ldots, n-1\}$ denoting the i-th update. Similarly to before, $\mathbf{X}_i^{d,te}$ denotes the dynamic features of a test sample at the i-th update and its size is $1 \times v$;
- $\boldsymbol{\pi}_i^{tr}$ is a matrix $m \times 3$, where 3 comes from having considered three classes (i.e. not-relevant, interesting and relevant). For $i = 0$ it is given by $\boldsymbol{\pi}_0^{tr} = \mathbf{J}_{m,1}\, p_0^{tr}$, where $\mathbf{J}_{m,1}$ is the all-ones matrix of size $m \times 1$ and p_0^{tr} is the vector 1×3 of prior probabilities of the training set. For $i \in \{1, \ldots, n-1\}$, it is defined by recursion as $\boldsymbol{\pi}_i^{tr} = \boldsymbol{\pi}_{i-1}^{tr} + \mathbf{y}_{i-1}^{tr}$, where $\mathbf{y}_i^{tr}$ is matrix $m \times 3$ containing the outputs of the learner M_i for all the training samples. Similarly to before, the use of apex te refers to a test sample;
- the learner M_i, already mentioned, is set up as depicted in panels B or C of Fig. 1, which will be detailed later. During the training process, its inputs are the labels $\mathbf{l}^{tr}$ and $\mathbf{X}_i^{tr} = [\mathbf{X}^{s,tr}, \mathbf{X}_i^{d,tr}, \boldsymbol{\pi}_i^{tr}]$; the latter is the matrix $m \times (u + v + 3)$ returned by the *Shared Representation Layer* block (SRL) where the features are concatenated by an early fusion approach. In the test phase, the input of M_i is only $\mathbf{X}_i^{te}$ provided by the SRL block;
- $\mathbf{y}_{n-1}^{te}$ is the vector 1×3 containing the output of the last learner, so that the final classification for a test sample is given by the max membership rule, i.e. $O^{te} = \text{argmax } \mathbf{y}_{n-1}^{te}$.

On this basis, Fig. 1 shows the proposed approach. Panel A represents the training and test phase, which consists of a chain of n learners M_i. In the training phase, each M_i is trained to predict the class labels of the training samples using past and current knowledge about the evolution of the interventions. Such knowledge is given by the shared representation provided by the SRL block, which includes static and dynamic features as well as $\boldsymbol{\pi}_i^{tr}$, a quantity that takes into account the past predictions of previous learners in the chain since an intervention can be assigned to different classes during its evolution.

In the test phase, the descriptors of each intervention feed all the learners of the chain, and the final label is given by the already mentioned quantity O^{te} provided by M_{n-1}. By construction, this approach takes into account all the information collected during the updates of the intervention as in the training phase.

A. ON THE STRUCTURE OF M_I

A first straightforward way to build M_i should consist of a binary classifier trained to distinguish between not-relevant and relevant classes, but, as mentioned at the end of section II, the introduction of a third virtual class may help the learning process. In this respect, Panel B of Fig. 1 shows that we appeal to a single multiclass classifier, whilst Panel C shows that M_i can be also organized in a cascade of binary classifiers. This approach is two-fold motivated: (i) in general discriminating between two classes is much easier than simultaneously distinguishing among more classes [9] and, in our context, the similarities between relevant and interesting samples could be better addressed by a dedicated classifier that follows a first one separating the not-relevant class from all the other instances; (ii) most of the available classification algorithms

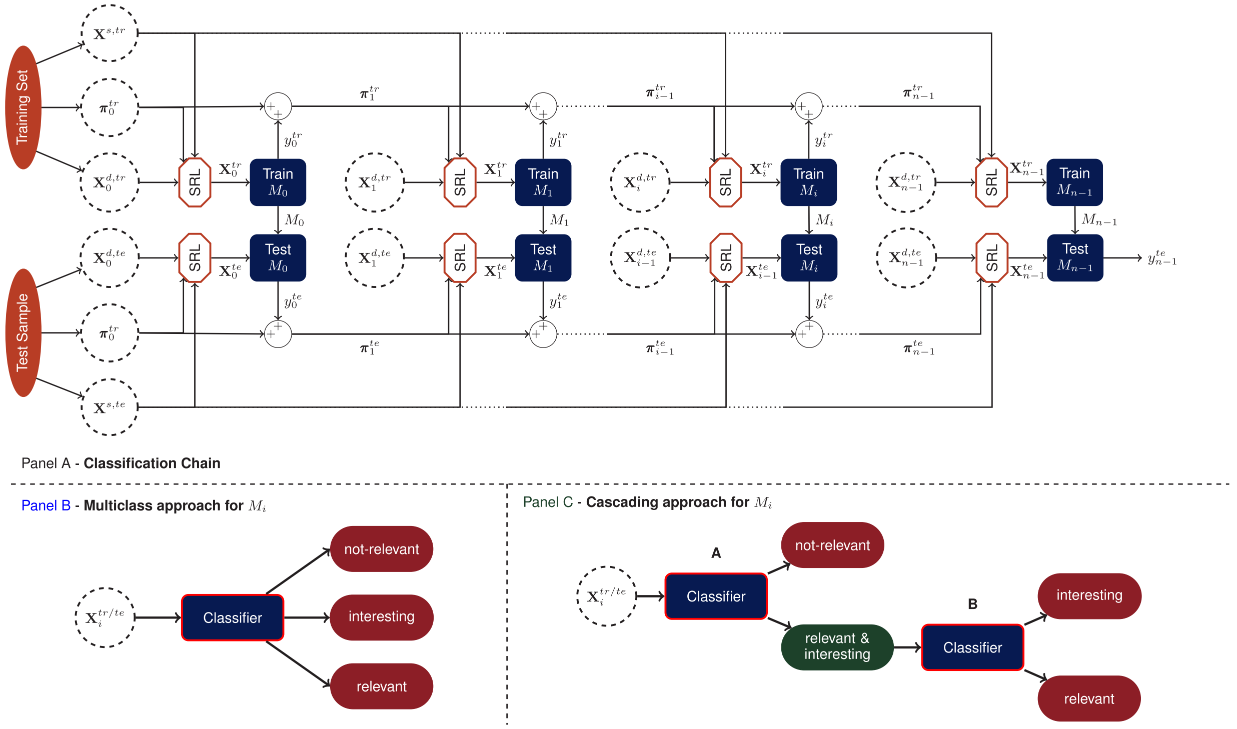


FIGURE 1. Panel A shows the classification pipeline of the method proposed. Panel B and C present the two classification approaches employed in the model train and test blocks of panel A.

are best suited to learn binary functions [10]. Note also that in the case of the cascade the quantity $\mathbf{y}_i^{tr/te}$ is given by the average of the outputs provided by the two classifiers if the prediction is relevant or interesting, otherwise it corresponds with the output of the first classifier. When comparing the results with those of the operator, however, the predictions are regrouped by merging the interesting samples with the not-relevant ones (see Fig. 5).

B. FEATURE COMPUTATION

As already mentioned at the beginning of the section, the features set is composed of *static* descriptors (Table 3), which do not change with the evolution of the event, and *dynamic* descriptors (Table 4), which do change and their computation has to be repeated for each of the updates considered. Table 5 shows an example of features computation for a relevant sample, starting from the CNVVF attributes to the features vector.

For the computation of the static features we both extract values from the CNVVF dataset records and employ data from other sources. In particular: we consider that the *month* and the *hour of the day* may influence the type of intervention required and possibly also how it evolves; the *description* attribute gives an indication of the type of event in progress; finally, the *time of first departure* may implicitly indicate the urgency of the intervention and the availability of resources at the time of the event. Furthermore, in order to have a statistical description of the geographical characteristics of the territory, we retrieve a publicly available dataset from *ISTAT*, the *National Institute of Statistics*, containing the five characteristics marked by an * in Table 3 [11].

Let us now turn our attention to the dynamic descriptors (Table 4). We identify four characteristics that represent the urgency and gravity of the event: the *update number* and the *event duration* make it possible to distinguish an event of short duration and with a reduced number of resources needed to deal with it from one that is protracted in time; the *average update time* may suggest the urgency of having more resources involved to address the event; finally, the *emergency vehicles* employed may indicate specific types of event. As an example of a dynamic feature computation, let's delve into this last descriptor: Table 5 the upper part reports that for each update we have two different types of vehicles employed for the rescue, i.e., the helicopter and the van for diving unit (column 4, rows 4-7). In the lower part of the same table there are the computed values for this descriptor, shown in two columns, one per type (columns 5-6, last four rows). The number reported in each row is computed considering the number of vehicles of a specific type that are out for the intervention at each update: for instance, in this intervention we have only one helicopter that is employed from the first update on, so we have 1 in each row of the corresponding column. Different it is the case of the van for diving unit: from the second update on, there is one more vehicle of this type that is employed for the rescue, so this number is incremented to finally reach 3, which is the total number of van for diving units that are out for this intervention in the last update.

TABLE 3. Descriptions of the static descriptors. Those marked with * are retrieved from the ISTAT dataset [11]. Each column of the dichotomous variables may assume a Boolean value.

Feature	Description	N° of columns	Encoding
*elevation zone**	It is the elevation of the geographical area of the event.	5	dichotomous
*centre altitude**	It is the altitude above sea level of the main town centre measured in meters.	1	standardised
*coastal municipality**	It is a dichotomous value indicating whether the municipality is coastal or not.	1	dichotomous
*mountain municipality**	It is a categorical value that distinguishes between a totally mountainous, partially mountainous or non-mountainous municipality.	3	dichotomous
*urbanisation degree**	It is a categorical value that distinguishes between a densely, moderately and slightly populated municipality.	3	dichotomous
month	the month of the emergency call, added to retrieve the seasonal component.	1	standardised
hour of the day	It is the hour the call was received and identifies the time of day when the event took place.	1	standardised
time of first departure	It is the elapsed time in minutes between the emergency call and the first departure of the emergency means; it is considered following the idea that a relevant intervention receives a quicker response than a not-relevant one.	1	standardised
description	It is one of the attributes of the CNVVF dataset and provides the reason for the intervention.	20	dichotomous

In addition, as already depicted at the beginning of the section, to transfer the information about past predictions along the chain, we design the features $\pi_i^{tr/te}$, named as *assignment probabilities*, initialized with the a priori probabilities of the three classes and updated each time summing ~~with~~ the probabilities returned by the i-th classifiers $y_i^{tr/te}$.

It is worth noting that *description* and *emergency vehicles* can assume a high number of different values, 65 and 75 respectively. While the latter is a list containing very heterogeneous vehicles that should not be grouped, in the former we identify redundant descriptions of the intervention reasons. This suggests we reduce such a high granularity by identifying 20 super-descriptions that group the 65 reasons for the interventions.

The data often do not have a suitable format to support the learning model, as it happens in our case. For this reason, we standardise to zero mean and unit variance the numerical features,[2] whilst we dichotomise the categorical ones. It should be noted that a simple dichotomisation does not allow us to distinguish events where more than one vehicle of the same type is involved. Therefore, instead of applying a pure one-hot-encoding, we report the number of vehicles involved in each category for these features. This information is shown in the last two columns of Tables 3 and 4.

IV. EXPERIMENTAL SETUP

The experiments are performed using the leave-one-out cross validation paradigm and we measure the performance by means of the following performance metrics: the accuracy, the recall, the precision and the F1-score.

Next sections explain how the depth of the chain is chosen and detail the classifiers employed, already mentioned in sections III and III-A.

A. ON THE CHAIN DEPTH

An important parameter of the proposed approach is the depth n of the classification chain. In order to identify the proper value for the task at hand we measure on a validation set drawn from the training set in 10-fold cross validation the performance of the approach for different updates in terms of recall and precision for $n \in \{0, \ldots, 9\}$,[3] and using three base classifiers belonging to different paradigms, which are described in section IV-B, for both the multiclass and the cascading approaches. For each depth value of the chain, in the recall-precision plane we estimate the distance of the measured performance scores from $(1, 1)$, which is the point corresponding to a perfect classification. This process enables us to plot a curve of how this distance varies with the number of updates considered, searching for a minimum: indeed the lower the distance from the optimum (1,1), the better the performance. To determine n we normalize the y-values of the curves in the $[0, 1]$ interval, and then we compute the mean and standard deviation of the 6 combinations of classifiers and classification approaches, as depicted in Fig. 2. As the update that reaches the lowest mean value is the sixth, we select it as the depth of the chain.

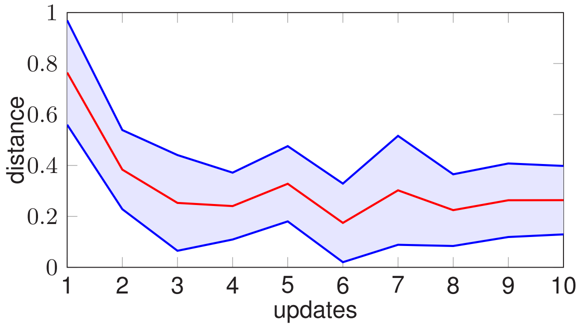


FIGURE 2. The figure shows the mean and standard deviation of the euclidean distances of the performance obtained in the relevant class recognition task of all possible combinations of classifiers and classification approaches from the optimum point in the recall-precision curve.

B. LEARNING PARADIGMS

We investigate learning paradigms belonging to different approaches. In particular, we use a support vector machine

[2] This operation is performed on the test set based on the parameters computed using the training set.

[3] The upper extreme can be eventually set to a different value, but in our case the preliminary analysis of the data suggested us to not proceed further.

TABLE 4. Descriptions of the dynamic descriptors.Each column of the dichotomous variables may assume a Boolean value.

Feature	Description	N° of columns	Encoding
update number	It is equal to the minimum between the number of the update under consideration and the maximum number of updates each event has.	1	standardised
event duration	If the event has a number of updates higher than the update under consideration, it is the difference in minutes between the *call time* and the *assignment time* (see Table 2) of the update following the one under consideration. Otherwise, it corresponds to the difference between the *call time* and the *closing time*.	1	standardised
average update time	For a given update, it is the ratio between the event duration at the beginning of the update (i.e. the event duration for the previous update), and the number of the given update itself. It is measured in minutes rounded to nearest integer. For events with only one update it is equal to the *time of first departure*.	1	standardised
emergency vehicles	It is the list of emergency vehicles that have been involved until the update under investigation.	75	dichotomous
assignment probabilities $\pi^{tr/te}$	They are the probabilities that the event belongs to the relevant, interesting and not-relevant classes. At the first update they are evaluated as the a priori probabilities obtained from the training set, and then they are adjusted by considering the assignment probabilities provided by the learning models at each update.	3	not-standardised

TABLE 5. The table reports an example of features computation for a relevant sample at its different updates: the upper-most part of the table shows the attributes of the CNVVF dataset divided into static and dynamic; in the middle part, it displays the message that was linked to the intervention. Finally, the bottom-most part shows the features vectors, once again divided into static and dynamic, for each single update obtained from the attributes given above.

Static CNVVF Attributes	*Intervention Date*	*Call Time*	*Road Name*	*Municipality*	*Province Abbreviation*	*Description*	*Typology Detail*	*Closing Date*	*Closing Time*
	Aug. 19, 2016	14:04	Palinuro	Centola	SA	Rescue of people	Rescue at sea	24/08/2016	19:11
Dynamic CNVVF Attributes	*Assignment Date*	*Assignment Time*	*Vehicle Type*						
update #1	19/08/2016	14:06	helicopter						
update #2	19/08/2016	16:11	van for diving unit						
update #3	20/08/2016	09:00	van for diving unit						
update #4	21/08/2016	09:07	van for diving unit						
SMS message	CON-VVF SEARCH FOR MISSING DIVERS (SA): The search continues for the three divers missing in a cave in the Faro di Palinuro area in the municipality of Centola. A further dive is underway to inspect the last stretch to an air bell at a depth of about 45 metres. At the moment, the search yielded negative results. VV.F. cave divers are on site from Lazio, Puglia and Veneto.								
Static Descriptors	*elevation zone*	*centre altitude*	*coastal municipality*	*mountain municipality*	*urbanisation degree*	*month*	*hour of the day*	*time of first departure*	*Description*
	4	336	1	TM	3	8	14	2	Rescue of people
Dynamic Descriptors	*update number*	*event duration*	*average update time*	*emergency vehicles*		*assignment probabilities*			
				helicopter	van for diving unit	not-relevant	interesting	relevant	
	1	127	2	1	0	0.53	0.16	0.31	
	2	1136	64	1	1	0.54	0.73	1.23	
	3	2583	379	1	2	0.69	1.29	2.02	
	4	7507	646	1	3	0.76	1.92	2.82	

(SVM) as a kernel machine, a random forest (RF) as an ensemble of tree classifiers and a Multilayer Perceptron (MLP) as a neural architecture. We use the default parameters values for all the classifiers because we are not interested in fine-tuning the models. Indeed, [12] empirically observes that in many cases the use of tuned parameters cannot significantly outperform the default values of a classifier suggested in the literature.

The support vector machine is built using the SVC model from the sklearn Python package [13]. It uses a radial basis function (RBF) kernel to deal with a nonlinear classification task, and a one-vs-one approach to tackle the multiclass classification problem, when needed.

The random forest is built using the RandomForestClassifier model from the sklearn Python package [13], its trees are built using the entropy as impurity measure and with a maximum depth of 10.

The multilayer perceptron is built using the sequential model from the Keras module, Python TensorFlow backend [14]. In its design we tried different values of layers and neurons per layer. Its final structure consists of five layers with 150, 100, 50, 30 neurons and in the last layer a number of neurons equal to the number of classes under consideration. Their activation functions are Sigmoid (layers I, II and IV), ReLU (layer III), Softmax (layer V). The network is trained up to 100 epochs with Adam optimiser, with a learning rate of 0.001, with categorical cross-entropy as loss function and an early-stopping criterion based on validation loss with a patience of 30 epochs. During training, 10% of the training data are used for validation.[4]

V. RESULTS

We now present the results achieved by the multiclass and the cascading approaches presented in section III and shown in Fig. 1. As already discussed in section IV-A, note that we analyze the approach by considering an increasing number of updates, up to 6.

The first row of Table 6 reports the performance computed considering the labels assigned by the firefighters' operators using the vademecum when he/she received the emergency call. It is worth noticing that in this scenario we regard as true positive the interventions marked as relevant that also received an SMS (so the relevant class for the proposed DSS), false positive the interventions marked as relevant that did not

[4] The code implemented in this work is available at: https://github.com/cosbidev/VVF_project

TABLE 6. Recognition results of the human operator (row 1) and of the proposed method, both for the multiclass (rows 2-4) and cascading (rows 5-7) approaches at the sixth update.

Method			Metric [%]			
			Accuracy	Recall	Precision	F1-score
Human operator			56.36	11.30	18.18	13.94
Learners	Multiclass	MLP	74.25	50.71	60.54	55.19
		RF	76.94	53.95	66.09	59.41
		SVM	76.46	47.46	67.61	55.77
	Cascading	MLP	73.23	57.91	57.10	57.50
		RF	76.15	55.93	63.46	59.46
		SVM	75.88	52.97	63.78	57.87

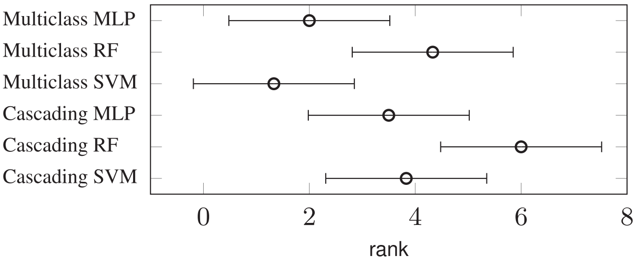


FIGURE 3. Rank of the various experiments computed with a multiple comparison test.

receive the SMS (so the interesting class for the proposed DSS), false negative the interventions that did receive an SMS and weren't marked as relevant by the operator and true negative all the rest.

Rows from 2 to 7 of the same table report the recognition results of the proposed approaches, both for the multiclass and the cascade, at the sixth update. Such performance show that, independently from the classifier employed, the proposed method has accuracies larger than the human operator. This particularly impacts the recognition performance of the relevant class (i.e. the recall) which are more than four times that of the operator. Similar considerations hold for the precision and, consequently, for the F1-score.

Let us now compare the results achieved by the different implementations of our proposal (rows 2-7 in Table 6). We notice that no one clearly outperforms the others, whilst the RF-based experiments show higher or at least more balanced performance, reflecting larger F1 values. This may be due to the ability of tree-based models to directly learn from both quantitative and categorical data, whereas the SVM and the MLP need for an encoding mechanism, which could bias the learning process.

To evaluate the statistical difference between the proposed methods, we perform a multiple comparison test using the statistics returned by Friedman's test performed on the F1-scores of the different experiments. Fig. 3 reports the ranks achieved by the different experiments where, on the one hand, we observe again that RF-based experiments get better performance and, on the other hand, multiclass SVM and multiclass MLP return F1-score statistically lower and different from the cascading RF ($p < 0.05$).

Let us now discuss how performance vary with the number of updates. In this respect, Fig. 4 shows that, in most of the cases, the F1-score increases with the number of updates. Continuous and dashed curves reveal that, except for the multiclass MLP, this happens whether we use a single multiclass classifier or the cascade of binary classifiers. As further proof of the fact that the cascading RF should be preferred to the others, in Fig. 4 we observe that the corresponding curve (dashed blue) reaches higher values than others at each update.

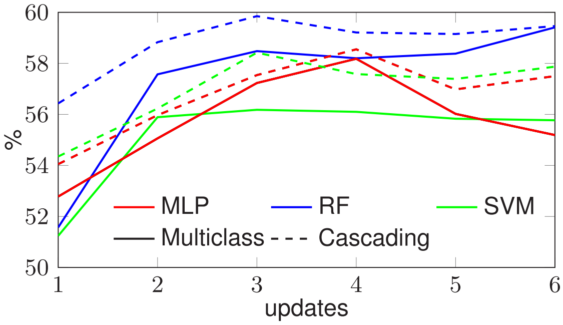
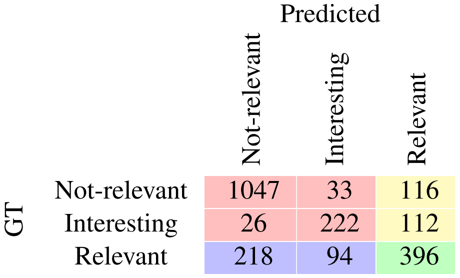


FIGURE 4. F1-scores of the tested methods varying the number of updates. Continuous and dashed lines refer to the multiclass and the cascading approaches, respectively.



		Predicted		
		Not-relevant	Interesting	Relevant
GT	Not-relevant	1047	33	116
	Interesting	26	222	112
	Relevant	218	94	396

FIGURE 5. Confusion matrix of the RF-based cascading approach. Red and green cells represent true negatives and true positives of the binary classification scenario, respectively. Indeed, not-relevant and interesting must collapse in a single class, i.e., the negative one. Yellow and purple cells correspond to false positive and false negative classifications, respectively.

As an additional test, we also verify that the proposed approaches, either multiclass or cascading, perform better than a pure binary one. Indeed, a binary RF reaches an F1-score equal to 56.50%, which is significantly lower than the cascading RF (Wilcoxon rank-sum test, $p < 0.05$). This suggests that the introduction of the interesting class does bring benefits in recognizing the class of relevant interventions. Note that some errors made in the classification of interesting interventions are tolerable since they are operators' mistakes. In fact, for the sake of completeness, Fig. 5 shows the confusion matrix of the RF-based cascading approach highlighting the results for the three classes. To compute the binary performance shown in Table 6 let us recall that interesting and not-relevant collapse in a single class, i.e., the negative one, as we highlighted in Fig. 5 using the same colour.

Given the imbalance of a-priori class probabilities in our dataset, we also attempt to introduce class weights in the RF,

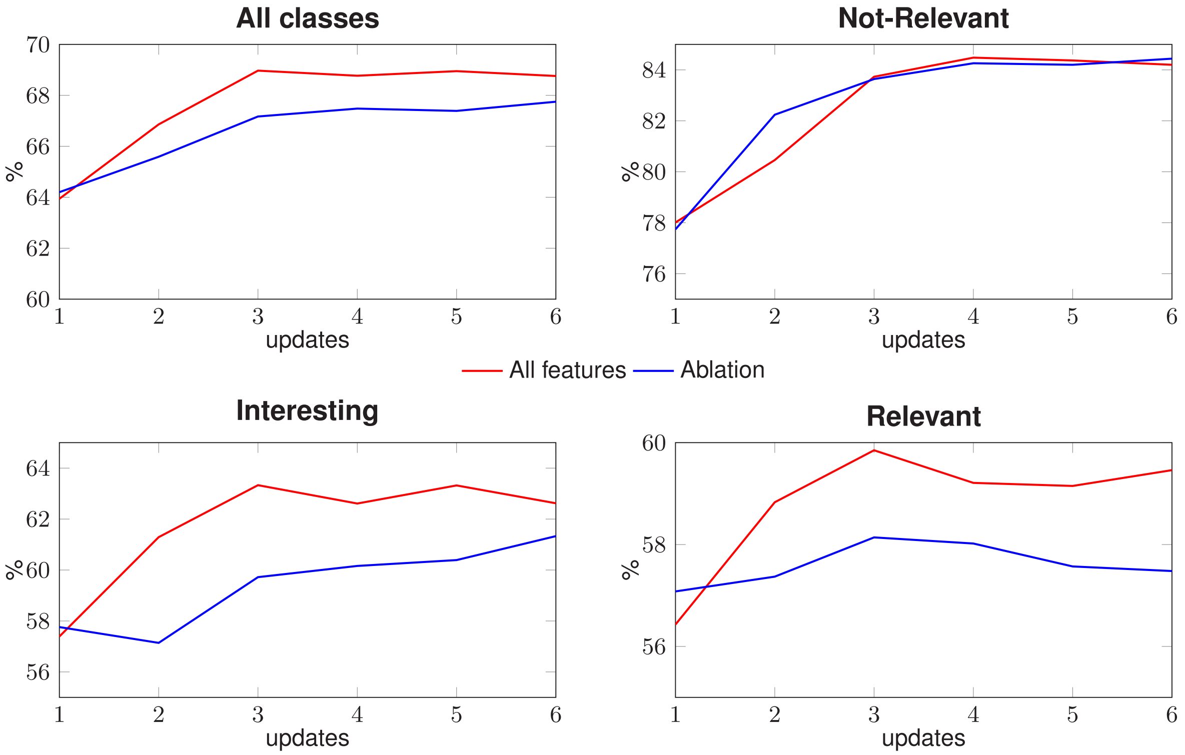


FIGURE 6. **Ablation test results in term of F1-score, reported per class.**

which is a standard approach to address this issue. We set the weights inversely proportional to class frequency in the training data. Note also that we consider only the RF as it is the model providing the best result. The model therefore becomes more liberal, as it labels more samples as positive. As usually happens in these situations [15], [16], this reflects in a larger recall (59.89%) and in a smaller specificity that, in turn, leads to a smaller accuracy (74.82%). However as the main limitation of the current vademecum-based approach is the large number of false positives, in the rest of this discussion we prefer to deepen the results attained by the more conservative configuration reported in Table 6, which does not adopt the class weights. We are also aware that there exists a large literature on more advanced methods to address class imbalance learning [15], but we deem that this investigation is out of the scope of this work.

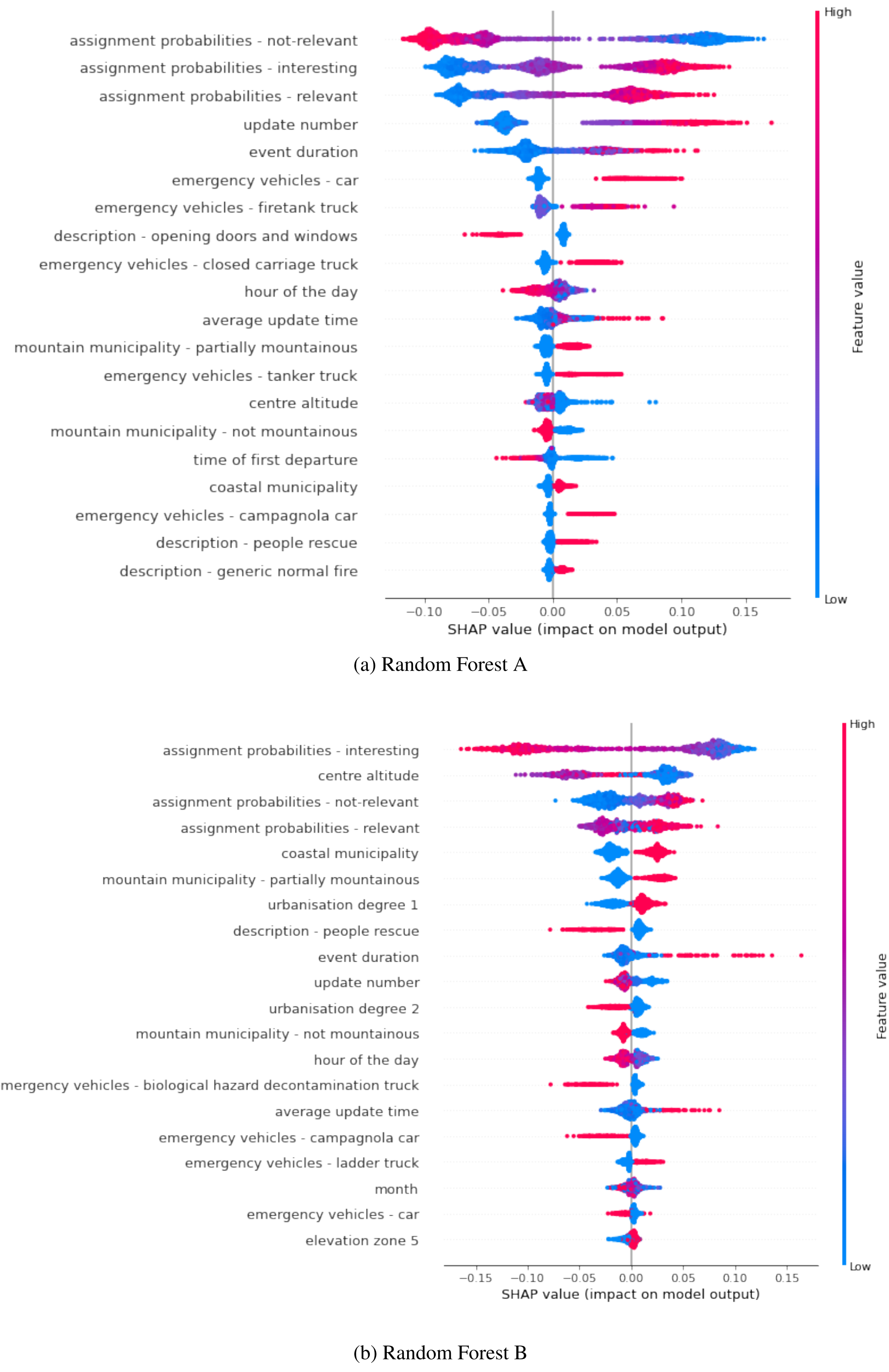
It is worth noting that all the results reported so far are obtained using the default parameters of the classifiers, as reported in section IV-B. Although we are not interested in fine-tuning the models according to [12], for the sake of completeness we investigate its utility on this dataset. To this end, we performed several tests changing the hyperparameters of the best model, i.e., the number of estimators $(50 - 100 - 150)$, the impurity measure (entropy and Gini) and the class weights (balanced or not) of the cascading RF, which is the best approach according to Table 6. Fine-tuning results agree with [12] and therefore confirm our initial hypothesis since the performance between the model with the default and the tuned parameters are not statistically different ($p > 0.1$).

As a further validation, the National Fire Corps provided us with another extraction of 265 samples. The labelling algorithm assigns 137 of them to the not-relevant class, 48 to the interesting one and the other 80 to the relevant class. Straightforwardly, when treated as a binary task, we have 185 and 80 not-relevant and relevant samples, respectively. The cascading RF trained on the dataset described in section III-B, when tested on this additional data, yields results consistent with those shown in Table 6. Indeed, it achieves an F1-score equal to 61%, whilst the human operator reaches 9%.

We also perform an ablation test to examine the effect on the predictions of the three assignment probabilities $\pi^{tr/te}$ (Table 4). To this end Fig. 6 shows that the F1-score obtained using all the features is larger than the corresponding value returned in the ablation test, both in the recognition of the relevant class and in the average recognition of all classes. We also perform a Wilcoxon rank-sum test revealing that such performance are statistically different ($p < 0.001$).

A. EXPLAINING FEATURES CONTRIBUTION

In recent years, besides developing learning algorithms able to cope with tasks of various natures, there has been a growing interest in explaining the predictions made by such models. Therefore several techniques that estimate the



(a) Random Forest A

(b) Random Forest B

FIGURE 7. SHAP summary plots of features' contributions in the RF-based cascading approach. Random forest A is the first in the cascade and it performs the classification between not-relevant and interesting/relevant, whilst random forest B is the second one and it distinguishes between interesting and relevant samples (as depicted in panel C of Fig. 1).

feature contributions to the model output are becoming more widespread.

Among the various approaches, we use the SHAP method: it is based on Shapley values, used in game theory to estimate the contribution of each player to the final payout [17]. With respect to the cascading RF, Fig. 7 shows the SHAP summary plots of the two learners used in the cascade, where the order of the features is based on their importance to distinguish the different types of interventions, i.e., the higher the position of a feature in the plot, the more important it is. Moreover, in order to understand the graphs, the colour of each dot indicates the value assumed by the sample according to the colour heatmap, whereas the feature contribution to the prediction of a specific class is represented by the extension of the points in a specific direction along the x-axis, i.e. the more the points stretch to the right the more the feature influences the prediction on the relevant class, on the contrary the more the points stretch towards the left the more the feature indicates the not-relevant class. For instance, in the first of the two graphs, high values of the feature *assignment probabilities - not-relevant* indicate the not-relevant class, whereas low values drive the prediction towards the relevant class.

The visual inspection of such plots suggests some observations: first, in both panels, we note that the *assignment probabilities* are the main features that allow the classifiers to distinguish the interventions being the topmost in the plots. Next, for classifier A, i.e., the first in the cascade distinguishing the not-relevant class from the other two, we notice that the temporal features *event duration* and *update number* have a great impact on the output of the model: this happens because relevant and interesting interventions are characterized by longer duration and a higher number of updates than the not-relevant ones. In classifier B we observe that relevant interventions are still distinguished by long duration, but with fewer updates compared to interesting ones. Further to this, the XAI analysis proves the hypothesis used to design the features and reported in section III-B: indeed, relevant interventions are characterized by low values of time of first departure compared to not-relevant interventions, and, being more prolonged over time, by high values of average update time. Finally, the XAI allows us to find out and understand which are some of the descriptions that misled the operator when assigning the event to the relevant class: for instance, looking at the plot for random forest A, the description *people rescue* seems to indicate a possible relevant intervention, but looking at the second graph (random forest B), we notice that this description is usually associated with interventions which are not truly relevant.

VI. CONCLUSION

A successful firefighters' intervention requires it to be rapid and appropriate to the event. There are a few works in the literature that use machine learning techniques to support firefighters' operations, and most of them aim at speeding up the intervention rather than making it appropriate to the event.

Within all the support requests arriving at the emergency department, the National Fire Corps needs to identify those that are related to relevant interventions to better allocate the available resources and alert the competent authorities. To this goal, we design a DSS that leverages an ad-hoc feature set, encompassing information collected during the emergency calls integrated with other information resources. The challenging problem we face is modelled as a binary task, which we decompose into a cascade of two dichotomies, a solution that is possible by introducing a third "virtual" class that stems from the analysis of the problem. Further to this, we boost our analysis leveraging the temporal evolution of each intervention with a chain architecture. Finally, we uncover the feature impact on the model output with explainability-based analysis.

The model performance significantly outperform the operators and show to be promising, suggesting that machine learning can help in automatically distinguishing relevant interventions.

Nevertheless we deem there is room for future work to improve the recognition performance of the relevant class, introducing for instance an ad-hoc feature selection stage or designing other features, which may take into account information about structures and activities around the event as well as data about past interventions at the same time or in surrounding locations.

Moreover, to further investigate the temporal evolution of interventions, some specific techniques could be used, such as Long Short Term Memory Neural Networks or Finite State Machines, as they might be able to extrapolate time-related information useful for the classification task.

AUTHORS' CONTRIBUTIONS

Camillo Maria Caruso: Conceptualization, Methodology, Software, Formal Analysis, Investigation, Writing-Original Draft, Writing-Review and Editing, Visualization; **Paolo Soda**: Conceptualization, Methodology, Formal Analysis, Writing-Review and Editing, Supervision; **Carlo Giammichele**: Resources, Data Curation, Writing-Review and Editing; **Francesca Rotilio**: Resources, Data Curation, Writing-Review and Editing; **Rosa Sicilia**: Conceptualization, Methodology, Formal Analysis, Writing-Review and Editing, Visualization, Supervision.

ACKNOWLEDGMENT

This work is based on data provided by the IT Services of Italian National Fire Corps. Camillo Maria Caruso is a PhD student enrolled in the National PhD in Artificial Intelligence, XXXVII cycle, course on Health and life sciences, organized by Università Campus Bio-Medico di Roma.

REFERENCES

[1] J. Weidinger, "What is known and what remains unexplored: A review of the firefighter information technologies literature," *Int. J. Disaster Risk Reduction*, vol. 78, Aug. 2022, Art. no. 103115.

[2] X. Lian, S. Melancon, J.-R. Presta, A. Reevesman, B. Spiering, and D. Woodbridge, "Scalable real-time prediction and analysis of San Francisco fire department response times," in *Proc. IEEE SmartWorld, Ubiquitous Intell. Comput., Adv. Trusted Comput., Scalable Comput. Commun., Cloud Big Data Comput., Internet People Smart City Innov.*, Aug. 2019, pp. 694–699.

[3] S. L. C. Ñahuis, C. Guyeux, H. H. Arcolezi, R. Couturier, G. Royer, and A. D. P. Lotufo, "Long short-term memory for predicting firemen interventions," in *Proc. 6th Int. Conf. Control, Decis. Inf. Technol. (CoDIT)*, Apr. 2019, pp. 1132–1137.

[4] C. Guyeux, J.-M. Nicod, C. Varnier, Z. A. Masry, N. Zerhouny, N. Omri, and G. Royer, "Firemen prediction by using neural networks: A real case study," in *Proc. SAI Intell. Syst. Conf.*, in Advances in Intelligent Systems and Computing, vol. 1037, 2020, pp. 541–552.

[5] S. Cerna, C. Guyeux, H. H. Arcolezi, and G. Royer, "Boosting methods for predicting firemen interventions," in *Proc. 11th Int. Conf. Inf. Commun. Syst. (ICICS)*, Apr. 2020, pp. 001–006.

[6] H. H. Arcolezi, J.-F. Couchot, S. Cerna, C. Guyeux, G. Royer, B. A. Bouna, and X. Xiao, "Forecasting the number of firefighter interventions per region with local-differential-privacy-based data," *Comput. Secur.*, vol. 96, Sep. 2020, Art. no. 101888.

[7] H. Arcolezi, S. Cerna, C. Guyeux, and J.-F. Couchot, "Preserving geo-indistinguishability of the emergency scene to predict ambulance response time," *Math. Comput. Appl.*, vol. 26, no. 3, p. 56, Aug. 2021.

[8] H. H. Arcolezi, S. Cerna, J.-F. Couchot, C. Guyeux, and A. Makhoul, "Privacy-preserving prediction of victim's mortality and their need for transportation to health facilities," *IEEE Trans. Ind. Informat.*, vol. 18, no. 8, pp. 5592–5599, Aug. 2022.

[9] S. Rajan and J. Ghosh, "An empirical comparison of hierarchical vs. two-level approaches to multiclass problems," in *Proc. Int. Workshop Multiple Classifier Syst.* Springer, Jun. 2004, pp. 283–292.

[10] T. G. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *J. Artif. Intell. Res.*, vol. 2, pp. 263–286, Jan. 1995.

[11] *Classificazioni Statistiche Comuni*, ISTAT, Rome, Italy, 2017.

[12] A. Arcuri and G. Fraser, "Parameter tuning or default values? An empirical investigation in search-based software engineering," *Empirical Softw. Eng.*, vol. 18, no. 3, pp. 594–623, Jun. 2013.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, A. Müller, J. Nothman, G. Louppe, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2012.

[14] M. Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: https://www.tensorflow.org/

[15] H. Kaur, H. S. Pannu, and A. K. Malhi, "A systematic review on imbalanced data challenges in machine learning: Applications and solutions," *ACM Comput. Surv.*, vol. 52, no. 4, pp. 1–36, Jul. 2020.

[16] C. Zhang, P. Soda, J. Bi, G. Fan, G. Almpanidis, S. García, and W. Ding, "An empirical study on the joint impact of feature selection and data resampling on imbalance classification," *Appl. Intell.*, vol. 2022, pp. 1–13, Jun. 2022.

[17] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "From local explanations to global understanding with explainable AI for trees," *Nature Mach. Intell.*, vol. 2, no. 1, pp. 56–67, Jan. 2020.

CAMILLO MARIA CARUSO received the B.S. degree in industrial engineering and the M.S. degree (Hons.) in biomedical engineering from Università Campus Bio-Medico di Roma, Rome, Italy, in 2018 and 2021, respectively, where he is currently pursuing the Ph.D. degree in artificial intelligence, XXXVII cycle, health and life sciences. His current research interests include deep learning, computer vision, and multimodal learning.

PAOLO SODA (Member, IEEE) received the degree (Hons.) in biomedical engineering and the Ph.D. degree in biomedical engineering (computer science area) from Università Campus Bio-Medico di Roma (UCBM), Rome, in 2004 and 2008, respectively. He is currently a Full Professor of computer science and computer engineering with UCBM. His research interests include artificial intelligence, pattern recognition, machine learning, big data analytics, data mining applied to data, signals, 2D and 3D images, and video processing and analysis.

CARLO GIAMMICHELE was born in Vasto, Italy, in 1961. He received the degree in electronic engineering from the University of Bologna, in 1993, and the Diploma degree in specialist in operative research and decision-making strategies. He is currently the Director of Information Technology Assistant Manager with the Information and Communication Technology Office, Department of Fire Brigade, Public Rescue and Civil Defense (IT Services of Italian National Fire Corps). He is a Technical Manager of the software development projects related to fire brigade rescue activities (CAD systems for fire fighters control rooms and asset management systems for CNVVF vehicles/equipments).

FRANCESCA ROTILIO received the degree (Hons.) in mathematics from the University of Rome "La Sapienza," in 1997, and the master's degree in quality of public administration from the University of Rome "Roma Tre," in 2006. She is currently the Director of Information Technology Assistant Manager of the Information and Communication Technology Office, Department of Fire Brigade, Public Rescue and Civil Defense (IT Services of Italian National Fire Corps). She is a Technical Manager with CNVVF, an institutional site, and a developer of information systems with business intelligent tools.

ROSA SICILIA received the degree (Hons.) in biomedical engineering and the Ph.D. degree in biomedical engineering (computer science area) from Università Campus Bio-Medico (UCBM), Rome, in 2016 and 2020, respectively, with a thesis on Micro-Level Rumour Detection on Twitter. After one year, she was a Postdoctoral Researcher. She is currently an Assistant Professor (RTDA) with UCBM, a position co-funded by Regione Lazio to focus on the project We-ease-it: A smart and intelligent outpatient clinic for Hospital 4.0. Her main research interests include machine learning and multimodal data mining fields.

Open Access funding provided by 'Università "Campus Bio-Medico" di Roma' within the CRUI CARE Agreement