Version of Record: https://www.sciencedirect.com/science/article/pii/S0926580516300966
Manuscript_9503d85849ab6757fbb651b2a92d8544

Application of Machine Learning to Construction Injury Prediction

Antoine J.-P. Tixier, Matthew R. Hallowell, Balaji Rajagopalan and Dean Bowman

ABSTRACT

The needs to ground construction safety-related decisions under uncertainty on knowledge extracted from objective, empirical data are pressing. Although construction research has considered Machine Learning (ML) for more than two decades, it had yet to be applied to safety concerns. We applied two state-of-the-art ML models, Random Forest (RF) and Stochastic Gradient Tree Boosting (SGTB), to a data set of carefully featured attributes and categorical safety outcomes, extracted from a large pool of textual construction injury reports via a highly accurate Natural Language Processing (NLP) tool developed by past research. The models can predict *injury type*, *energy type*, and *body part* with high skill ($0.236<RPSS<0.436$), outperforming the parametric models found in the literature by a factor. The high predictive skill reached suggests that injuries do not occur at random, and that therefore construction safety should be studied empirically and quantitatively rather than strictly being approached through the analysis of subjective data, expert opinion, and with a regulatory and managerial perspective. This opens the gate to a new research field, where construction safety is considered an empirically grounded quantitative science. Finally, the absence of predictive skill for the output variable *injury severity* suggests that unlike other safety outcomes, *injury severity* is mainly random, or that extra layers of predictive information should be used in making predictions, like the energy level in the environment. In the context of construction safety analysis, this study makes important strides in that the results provide reliable probabilistic forecasts of likely outcomes should an accident occur, and show great potential for integration with building information modeling and work packaging due to the binary and physical nature of the input variables. Such data-driven predictions had been absent from the field since its inception.

1. INTRODUCTION AND MOTIVATION

Construction is one of the largest industries in the United States, but is also one of the deadliest (Bureau of Labor Statistics 2013). Between 1992 and 2010, an average of 730 lives have been claimed each year (CPWR 2013). Despite the numerous efforts that have been motivated by this alarmingly poor performance, injury statistics have not significantly improved in the past decade (BLS 2013). This might be explained by the fact that the construction industry has reached saturation with respect to traditional approaches to safety and that innovations are needed (Esmaeili and Hallowell 2011a). Risk analysis has emerged as a promising alternative to managerial and regulation-based approaches. However, construction safety risk analyses are currently limited because existing techniques overlook the complex and dynamic nature of construction sites and are not based on empirical data.

To jointly address these limitations, Esmaeili and Hallowell (2012, 2011b) laid the groundwork of a new conceptual framework, offering a systematic and comprehensive way to extract safety critical structured information from unstructured injury reports. Unlike traditional safety risk analysis techniques, this attribute-based approach renders construction injuries as the resulting outcome of the joint presence of a worker and the interplay among a finite set of universal descriptors of the work environment that are observable before an injury occurs. These binary attributes, also called injury precursors, make physical sense and are related to construction means and methods, human behavior, and environmental conditions. For instance, in the following excerpt of an injury report: “employee was welding and grinding inside tank and experienced discomfort to left eye”, four fundamental attributes can be identified: (1) *welding*, (2) *grinding*, (3) *tank*, and (4) *confined workspace*.

The attribute-based framework derives its strength from its ability to capture and encode the information of every possible construction situation in a finite, standardized format, regardless of trade, project type, or industry sector. Therefore, as illustrated in Figure 1, extracting attributes and various safety outcomes from injury reports (i.e., objective empirical data) enables the constitution of a structured, consistent multivariate data set ideally suited for data mining, predictive modeling, and, thus, for knowledge discovery. Such new knowledge can enhance understanding of the underlying mechanisms that shape construction safety risk and create injuries. More precisely, *this study seeks to demonstrate that the workflow illustrated in Figure 1 is viable and can be used to produce empirically-driven models with high predictive skill.* A fundamental postulate made here is that construction safety is not a strictly managerial outcome, but rather features a non-random component that can be studied by means of observation, like any other natural phenomenon. If this assumption holds, adopting the attribute-based framework would succeed in transforming construction safety research from opinion-based and qualitative to objective, empirically grounded quantitative science.

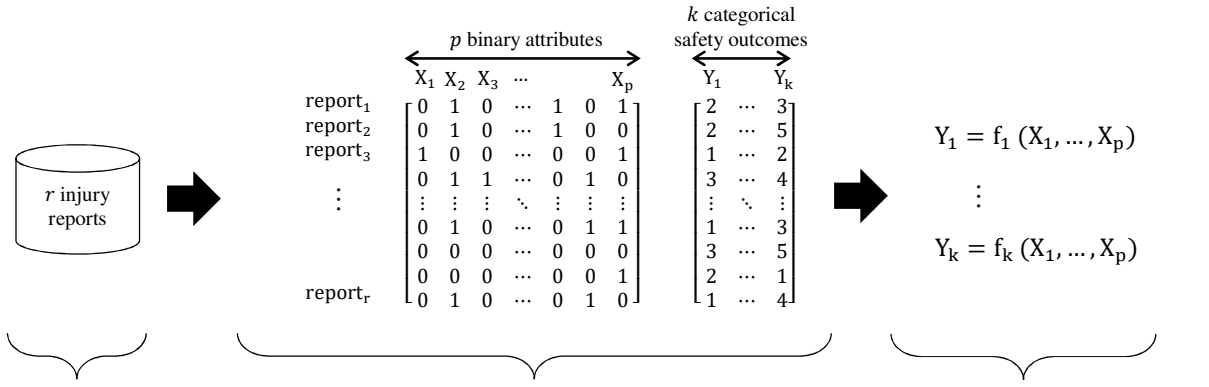

Figure 1. The derivation of predictive models from injury reports is enabled by the attribute-based framework

The effectiveness of the attribute-based framework depends on a number of methodological parameters including: (1) the way attributes are created and defined, (2) the quality and quantity of the injury reports available, (3) the technique with which attributes are extracted from the reports, and (4) the methods used for data mining and predictive modeling. As will be discussed in the background section, all previous work in this emerging research area (e.g., Esmaeili et al. 2015a, Esmaeili et al. 2015b, Prades 2014, Desvignes 2014, Esmaeili and Hallowell 2012, 2011b) is subject to limitations with respect to one or more of the aforementioned parameters.

Building on three recent studies (Prades 2014, Desvignes 2014, Tixier et al 2016) that respectively addressed the limitations pertaining to the first three of the aforementioned criteria, here we tackle the limitations related to the fourth: predictive modeling. More specifically, two state-of-the-art machine learning (ML) algorithms, Random Forest (RF) and Stochastic Gradient Tree Boosting (SGTB), were used to predict safety outcomes from fundamental construction attributes. As will be shown, the models built outperform that of past research, in terms of predictive skill, variety of outcomes predicted, and actionable feedback that can be used to direct efforts towards targeted preventive actions and corrective measures.

2. BACKGROUND AND POINT OF DEPARTURE

This section provides the inspiration for our work, a brief description of past research in the domain of attribute-based safety analysis and in the application of machine learning in the construction industry, and the expected contributions.

2.1. Why does prediction of safety outcome matter?

Many industries, including construction, struggle with decision-making under uncertainty. Making the wrong decisions can have dramatic consequences, especially since lives are at stake. In healthcare, for example, Seera and Lim (2014) observed that lack of experience, information overload, and unawareness of the most recent advancements in medical research were the leading causes of misdiagnosis by physicians. In the exact same way, even an experienced construction worker or safety manager has limited personal history with accidents. They may have witnessed, in their entire professional life, hundreds of near misses and first aid injuries, dozens of medical cases and lost work time injuries, and, perhaps, a few permanent disablement injuries and fatalities. Because of this limited experience with incidents, they may misdiagnose the risk of a given construction situation. It is well known that poor hazard recognition skill is a proximal cause of risk misperception and injury in construction (Albert et al. 2014, Carter and Smith 2006). People working upstream of the construction phase, like designers, face an even greater risk of failing to recognize hazards and misestimating risk (Albert et al. 2014, Almén and Larsson 2012).

Furthermore, without even considering the limited experience problem, human judgment and intuition will always be subject to important biases and fallacies (e.g., Tversky and Kahneman 1981). Also, humans have very limited capability of inducing knowledge from large numbers of observations (Skibniewski et al. 1997). This is due to the fact that human short-term memory is only capable of handling at most seven items evaluated for seven attributes at the same time (Miller 1956).

On the other hand, ML can induce general rules from very large amounts of cases belonging to highly dimensional spaces, and is therefore a way to ground safety-related decisions under uncertainty on empirical knowledge. This could lead to improved decision-making and save lives. Indeed, other industries have begun to realize great benefits by transitioning from subjective to objective decision making thanks to statistical learning. For instance, Seera ad Lim (2014) trained ML models on large numbers of health records to automatically diagnose new patients, providing physicians with an opportunity to reconsider initial decisions and improve diagnosis accuracy.

2.2. Limitations of previous work on attribute-based construction safety

Although Esmaeili and Hallowell (2012, 2011b) made important strides by introducing and using the attribute-based framework for the first time, some serious limitations remained. In particular, some of the attributes identified via manual content analysis were not in full accordance with the framework as they were outcomes (e.g., *structure collapse*, *falling from roof*). By nature, an injury precursor should be observable *before* an injury occurs. Some other attributes were overlapping (e.g., *working underground*, *working in a confined space*), or loosely defined (e.g., *not considering safety during site layout*). Finally, the content analysis had rather low consistency (76% of inter-coder agreement), and only 300 reports all related to high severity struck-by injuries were analyzed, so only part of the picture was captured.

Esmaeili et al. (2015a) took the research a step further by using commercial software to automatically extract attributes from a larger amount of reports (1,450). However, the low accuracy of the procedure (21% disagreement between manual and automated coding on average) was a significant limitation, as it compromised the reliability of the data set obtained. In addition, the usefulness of the models built was restricted by the fact that only high severity struck-by injuries were taken into account. It should also be noted that only 22 attributes were considered.

Finally, Esmaeili et al. (2015b) used the data set obtained by Esmaeili et al. (2015a) to predict a binary severity outcome (fatality/no fatality) via a logistic regression model taking principal component scores as input variables. On the full training data set, the best model obtained a Rank Probability Skill Score (RPSS) of 0.116, which indicates modest skill (Goddard et al. 2003). In addition, this score was an overly optimistic estimate of the true predictive skill, as the model was tested on the very same observations that were used for training. To ensure unbiased estimation of a model's true ability to extrapolate, testing should always be conducted against unseen observations, using a separate test set when there is enough data, or cross-validation else (Hastie et al. 2009, pp. 222-223). Another limitation of Esmaeili et al. (2015b) is the use of logistic regression, a parametric, linear and global model which is by definition unable to capture the nonlinear and local relationships that may exist among predictors and targets (Towler et al. 2010, Rajagopalan et al. 2005). Also, because these relationships are unknown, parametric models are not best suited for skillful prediction.

To address the abovementioned limitations, we first used a broadened and more robust list of 80 attributes engineered and validated by a team of 8 researchers (Prades 2014, Desvignes 2014) and slightly modified by Tixier et al. (2016). This list is provided in Table 2. Second, we used a rather large database of 5,298 injury reports that featured all types of injuries and was representative of the true distribution of injury severity. Third, a large and reliable data set of attributes and outcomes was automatically extracted from the database of injury reports by a highly accurate (96% in F1 score) natural language processing (NLP) program developed by Tixier et al. (2016), ensuring high data quality. Finally, we used RF and SGTB, two cutting edge statistical learning algorithms, to predict safety outcomes from attributes with high skill. Since RF and SGTB both use decision trees as their base models, these two techniques can capture both nonlinear and linear; local and global relationships between input and output variables.

2.3. Previous used of machine learning in construction

Construction research has considered ML for more than two decades. Moselhi et al. (1991) first discussed the potential applications of neural networks in construction engineering and management and developed a prototype providing optimum markup estimates from attributes describing bid situations, such as the number of competitors or the contractor's estimated cost. Later, Skibniewski et al. (1997) applied the AQ15 algorithm on a collection of 31 training examples to automatically learn the mapping between constructability (poor, good, excellent) and 7 predictors, such as the reinforcement ratio of the beam and the number of walls attached to it. Soibelman and Kim (2002) applied decision trees and neural networks to a construction management database to identify the causes of delays.

More recently, Lam et al. (2009) found that support vector machines could produce accurate forecasts of contractor prequalification using input variables such as financial strength, current workload, quality management, and environment, health and safety considerations. Also, Cheng et al. (2011, 2010) used a support vector machine optimized via a fast messy genetic algorithm to estimate building cost and loss risk from ten input variables, such as change orders and number of rainy days, and to estimate the loss risk associated with a given project given project duration, number of floors, construction season, and geological conditions. Finally, Yang et al. (2010) developed an algorithm to automatically track workers in digital videos; Tsanas and Xifara (2012) used RF to predict heating and cooling loads of residential buildings from wall area, glazing area, overall height, and other input variables; and Son et al. (2012) used a support vector machine model to detect concrete structural components in color images from actual construction sites.

Although not exhaustive, this short review of the literature shows that ML has a quite long history of being used in construction research for a variety of applications. However, to the best of our knowledge, this is only the second time that supervised learning algorithms are used to predict construction safety-related outcomes from empirical data (after Esmaeili et al. 2015b).

2.4. Goal of this study

The goal of the present research effort is to apply Random Forest (RF) and Stochastic Gradient Tree Boosting (SGTB), two widely used and highly successful Machine Learning (ML) algorithms, to attribute and outcome data extracted from a large body of injury reports. Note that we could have included other supervised classifiers in our comparison, like Support Vector Machines or Neural Networks, but we were mainly interested in testing whether fundamental construction attributes could make good features at all, not about extensively testing all major classification algorithms. The predictive models obtained can be used to augment the experience of construction professionals with lessons learned from empirical data representing millions of worker-hours, far exceeding the exposure of even the largest and most experienced group of experts. This extensive amount of empirical knowledge can be used with profit to improve safety management in the design, work packaging, and execution phases of a construction project.

In practice, the models developed assign a probability of occurrence to each level of each safety outcome from a simple description of the work environment in terms of attributes. An example is given in Figure 2 for the safety outcome *body part injured.* Such probabilistic forecasts provide some insight as to which preventive and/or corrective actions to take, allowing for better-informed, safer proactive decision-making. Providing a risk estimate (green, orange, red) for a given combination of observed attributes such as in Prades (2014) is useful, but predicting the most likely categories of various safety outcomes is a complementary and equally valuable strategy.

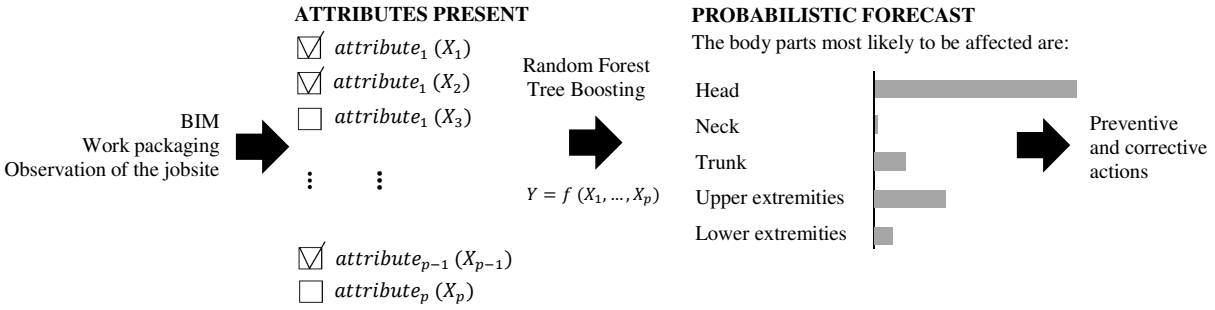

Figure 2. Practical use of the predictive models built in this study

2.5. Characteristics of the data set

We had access to a raw database of 5,298 injury reports gathered from more than 470 contractors involved in industrial, energy, infrastructure, and mining work throughout the world and representing millions of worker-hours. More details about these data can be found in Prades (2014), Desvignes (2014), and Tixier et al. (2016). These reports were automatically scanned for the attributes shown in Table 1 and the safety outcomes listed in Table 2 by Tixier et al.'s (2016) NLP system.

As summarized in Table 2, the safety outcomes predicted in this study were the (1) *type of energy* involved in the accident, (2) *injury type*, (3) *body part* affected, and (4) *injury severity*. The outcome *energy type* was taken into account based on the theory that any injury can be associated with the release of some form of energy (Fleming 2009, Haddon 1973). For *injury type*, *body part*, and *injury severity*, the classification scheme is consistent with that of the Bureau of Labor Statistics (BLS 2010) and the Occupational Safety and Health Administration (OSHA) (BLS 2010, Hallowell 2008).

Table 1. Eighty context-free validated injury precursors from Tixier et al. (2016)

UPSTREAM*	*n*	Rebar	155	Screw	37
Cable tray	48	Scaffold	300	Slag	75
Cable	75	Soffit	12	Spark	9
Chipping	34	Spool	52	Slippery surface	142
Concrete liquid	58	Stairs	137	Small particle	401
Concrete	165	Steel sections	759	Adverse low temperatures	123
Conduit	56	Stripping	114	Unpowered tool	611
Confined workspace	129	Tank	85	Unstable support/surface	8
Congested workspace	13	Unpowered transporter	53	Wind	109
Crane	69	Valve	79	Wrench	110
Door	85	Welding	200	Lifting/pulling/manual handling	553
Dunnage	29	Wire	131	Light vehicle	133
Electricity	3	Working at height	268	Exiting/transitioning	132
Formwork	143	Working below elevated workspace/material	50	Sharp edge	47
Grinding	133	Drill	97	Splinter/sliver	41
Grout	18	**TRANSITIONAL**		Repetitive motion	66
Guardrail/handrail	91	Bolt	186	Working overhead	14
Heat source	111	Cleaning	119	**DOWNSTREAM**	
Heavy material/tool	79	Forklift	39	Improper body position	88
Heavy vehicle	143	Hammer	149	Improper procedure/inattention	57
Job trailer	24	Hand size pieces	172	Improper security of materials	87
Lumber	252	Hazardous substance	156	Improper security of tools	28
Machinery	189	Hose	95	No/improper PPE	23
Manlift	66	Insect	105	Object on the floor	174
Stud	31	Ladder	163	Poor housekeeping	2
Object at height	86	Mud	35	Poor visibility	12
Piping	388	Nail	94	Uneven surface	59
Pontoon	15	Powered tool	239		

* Upstream precursors can be anticipated as soon as during the design phase; transitional precursors are generally not identifiable by designers but can be detected before construction begins based on knowledge of construction means and methods; and downstream precursors are mostly related to human behavior and can only be observed during the construction phase. Note that the original list of attributes is due to Desvignes (2014), but minor modifications were made by Tixier et al. (2016).

Table 2. Safety outcomes predicted

ENERGY SOURCE	**INJURY TYPE**	**BODY PART**	**INJURY SEVERITY**
Biological	Caught in or compressed	Head	Pain
Chemical	Exposure to harmful substance	Neck	First aid
Electricity	Fall on same level	Trunk	Medical case
Gravity	Fall to lower level	Upper extremities	Lost work time
Mechanical	Overexertion	Lower extremities	Permanent disablement
Motion	Struck by or against		Fatality
Pressure	Transportation accident		
Radiation			
Thermal			

It should be noted that Prades (2014) and Desvignes (2014) ensured the validity and relevance of the attributes created via content analysis by adhering to a strict coding scheme, implementing an iterative process with team-based calibration meetings, and using peer reviews and random checks by external reviewers with a stringent 95% agreement threshold. Such great care was taken because this procedure, called *feature engineering*, is of paramount importance to ML success (Domingos 2012). Tixier et al.

(2016) also tuned their NLP tool by adopting an iterative process involving at each step careful reviews by 7 researchers of 140 randomly selected reports scanned by the system. At each round, lessons learned from examining the errors made by the tool were used to improve skill. A harsh 95% threshold in accuracy was exceeded after 4 iterations (96%). In particular, the NLP system attained precision and recall rates of 95% and 97% for attributes, and error rates of 5.7% for both *energy type* and *injury code*. The NLP tool was designed to return "not detectable" when multiple body parts are detected in a given report, or when the information is missing. However, on the 93.75% of reports it could label, the tool proved 100% accurate (Tixier et al. 2016).

900 reports out of the 5,298 available were not associated with any attribute, and were therefore removed. An inspection of these reports showed that they were very short and did not contain any attribute-related information. The attributes *poor housekeeping* and *electricity* were discarded due to their absolute rarity (2 and 3 observations only), as well as the energy type *electricity* (3), and the injury types *transportation accident* (4) and *fall to lower level* (18). This made for a final data set of $r = 4{,}398$ observations, $p = 78$ attributes, and $k = 4$ safety outcomes (using the notation from Figure 1). The number of times each attribute appeared in this data set are shown in Table 2. The safety outcome *body part affected* could not be inferred for 831 reports, so for this particular target, only 3,556 observations were available for training. Also, because it requires mental projection, Tixier et al.'s (2016) NLP tool cannot extract the safety outcome *injury severity*, so for this prediction task, the 1,829 reports manually analyzed by Prades (2014) and Desvignes (2014) had to be used. Finally, the levels *permanent disablement* and *fatality* were removed (respectively one and no observation), and *pain* (159 observations) was combined with *first aid* (1,362) since the difference between these two severity levels appeared to be very tenuous. The counts of each category of the safety outcomes in the final data sets are presented in Table 3.

Table 3. Number of observations for each level of the four safety outcomes predicted

Energy source	***n***	**Injury type**	***n***	**Body part**	***n***	**Severity**	***n***
Biological	108	Caught in or compressed	334	Head	899	Pain/First aid	1,521
Chemical	197	Exposure to harmful substance	496	Neck	61	Medical case	206
Gravity	1,030	Fall on same level	570	Trunk	354	Lost work time	101
Mechanical	74	Overexertion	594	Upper extremities	1532	TOTAL	1,828
Motion	2,780	Struck by or against	2,401	Lower extremities	710		
Pressure	47	TOTAL	4,395	TOTAL	3556		
Thermal	151						
TOTAL	4,387						

As one can see from Table 3, four multi-class prediction tasks were to be tackled in this study (i.e., there were four categorical safety outcomes to predict). Using the notation from Figure 1, the four output variables were $Y_1 = energy\ source$ (7 levels), $Y_2 = injury\ type$ (5 levels), $Y_3 = body\ part$ (5 levels), and $Y_4 = injury\ severity$ (3 levels). For each safety outcome (i.e., each Y_k), the goal was to determine the best f_k such that $Y_k = f_k(X_1, \dots, X_p)$, where $(X_1, \dots, X_p)$ are the fundamental construction attributes presented in Table 2. The methods used and procedure followed to accomplish these tasks are presented next.

3. APPLICATION OF MACHINE LEARNING

We used the $r = 4{,}398$ by $p = 78$ structured data set of attributes and outcomes shown in Figure 1 ($p = 78$ since *poor housekeeping* and *electricity* were removed as previously explained). The features, or input variables, were the fundamental construction attributes $(X_1, \dots, X_{78})$ listed in Table 2, such as *welding*, *uneven surface*, or *adverse low temperatures*, and the targets, or output variables, were the four categorical safety outcomes $(Y_1, \dots, Y_4)$, listed in Table 3: *energy type*, *injury type*, *body part*, and *injury severity*. Each injury report, also referred to as an observation or training example in what follows,

associated a specific combination of attributes to a specific combination of safety outcomes. Based on such training data, ML algorithms could infer rules mapping combinations of attributes to levels of safety outcomes, and use these rules later on to predict the most likely outcomes for brand new observations.

ML was preferred over parametric modeling because the latter is not optimal when little knowledge is available about the phenomenon studied. Indeed, parametric modeling imposes a model *a priori* to the data, either arbitrarily or based on some knowledge about the underlying process. Therefore, if the model selected is a poor representation of the phenomenon studied in the first place, it may be nothing more than "the right answer to the wrong question" (Breiman 2001a). On the other hand, ML algorithms do not assume that the data have been generated by any parametric model prescribed by the user. Rather, the assumption is that independent and dependent variables are related in a totally complex and unknown manner. Both linear and nonlinear relationships can be captured, as well as complex high-order interactions among variables, without imposing any formal model and its inherent suite of limitations.

More specifically, we used Random Forest (RF) and Stochastic Gradient Tree Boosting (SGTB) as our machine learning (ML) algorithms. These two techniques were used as they currently stand among the most popular and successful supervised machine learning (ML) methods available. The rationale for using two different algorithms stemmed from (1) the exploratory nature of this research, (2) the absence of general rule saying that SGTB is always better than RF and vice versa (performance really depends on the data and on the problem at hand), and (3) the interest in comparing predictive skill. As already remarked, we could have included other ML models in our comparison like Support Vector Machines or Neural Networks, but we were mainly interested in testing the extent to which fundamental construction attributes would carry predictive skill, independently of the classification algorithm. After briefly introducing RF and SGTB, we present and justify the methodological choices made to address class imbalance and parameter optimization, and discuss the application of the procedures in practice.

3.1. Random Forest (RF)

The RF algorithm Breiman (2001b) grows many decision trees built via CART (Breiman et al. 1984) and aggregates their output (majority vote here, in the case of classification). Using binary splits, decision trees recursively partition the predictor space by identifying the regions that have the most homogeneous responses to predictors (Elith et al. 2008). Then, a constant is locally fit to each final region (or leaf): for a categorical outcome variable, it is the most probable category. As opposed to *global* models such as logistic regression, where the same equation holds over the entire data space, trees are *local* models, enabling them to adapt to and truly represent the multiple domain-specific facets of the relationships between input and output variables. RF inherits many of the advantages of trees, such as the ability to capture complex nonlinear high-order interactions among predictors, to handle highly dimensional data sets with large numbers of observations, and the robustness to outliers and to the inclusion of irrelevant predictors (Sutton 2005, Timofeev 2004). Furthermore, by growing each tree on randomly selected observations (with replacement) from the original data set, and by only trying a random subset of the input variables at each split, RF achieves much greater predictive accuracy than a single tree.

RF was selected because it stands among the most accurate general-purpose classifiers to date (Biau 2012), and has shown to be effective in a variety of other fields. To cite only a few examples, the RF algorithm has been used with success to predict patient risk for various diseases (Lebedev et al. 2014, Khalilia et al. 2011), identify central genes (Díaz-Uriarte and de Andrés 2005), develop automated stock trading strategies (Booth et al. 2014), forecast air traffic delays (Rebollo and Balakrishnan 2014), analyze the risk of mortgage prepayment (Liang and Lin 2014), determine the likelihood that a customer will cease doing business with a company (Xie et al. 2009), predict horse race outcomes (Lessmann et al. 2010), and to evaluate the likelihood of being elected to the baseball hall of fame (Freiman 2010).

The tuning parameters of RF are the number $ntree$ of trees in the forest, and the number $mtry$ of predictors randomly considered as candidates at each split. The "randomForest" package (Liaw and Wiener 2002) of the R programming language (R Core Team 2015) was used in this study to build all the RF models.

3.2. Stochastic Gradient Tree Boosting (SGTB)

Like RF, Boosting is an ensemble approach that combines many base models and let them vote to generate forecasts (Freund et al. 1999). Because it can turn an ensemble of weak classifiers (each only slightly better than random guessing) into a strong classifier, Boosting was qualified as being one of the most powerful advances in ML in the last 20 years (Hastie et al. 2009, p. 337). Like RF, Boosting is often used with decision trees as base models, as it has proven extremely effective in that case (Hastie et al. 2009, p. 340). However, while RF grows large trees in parallel, Tree Boosting builds a sequence of very small trees, such that each successive tree focuses on capturing the regions of the training set that were missed by the preceding one.

SGTB (Friedman 2002, 2001) is an improvement of Tree Boosting where the gradient of some differentiable loss function is used to identify the regions missed, and a random subsample of the training set (instead of the full training set) is used to fit and add each new tree to the model. Some examples of loss functions are the squared error (for regression), or multinomial deviance (used here, for classification). In this study, SGTB models were created with the "gbm" R package (Ridgeway et al. 2015).

SGTB has five tuning parameters. The first is the number $n.tree$ of trees in the sequence. A high number of trees is needed to achieve good learning, but unlike with RF, too many trees can lead to overfitting on noisy data sets (Opitz and Maclin 1999), so close monitoring of $n.tree$ is indispensable. Overfitting describes the instance when a too complex model encodes the peculiarities of the training data (i.e., the noise) as rules rather than its general structure (i.e., the signal). It always deteriorates extrapolation. The second parameter of Boosting is the size of the trees, which is controlled by $interaction.depth$. This parameter is very important, as it defines the order of predictor-predictor interaction that can be captured. For instance, specifying trees with two final nodes (one single split) allows only main effects to be modeled. Trees with three final nodes (two splits) allow first-order (two-variable) interactions to be captured, and so forth (Hastie et al. 2009, p. 362). The third parameter is the $learning.rate$, which is a factor between 0 and 1 that shrinks the contribution of each new tree added in the series. By delaying the point when overfitting is reached, low values of $learning.rate$ (<0.1) allow more trees to be added to the sequence, which dramatically improves performance (Friedman 2001). The fourth parameter is the minimum number $n.min$ of observations allowed per node. Larger values of $n.min$ generate smaller trees, which are less sensitive to noise. The proportion of training examples randomly drawn at each round is the fifth and last tuning parameter, called the $bag.fraction$.

3.3. Class imbalance issue

Our data set featured some significantly underrepresented categories, which is a well-known issue in areas like gene profiling, credit card default, or fraud detection (Tang et al. 2009, Jaehee and Thon 2006, Chawla et al. 2002). Learning from such data sets is a challenge for all ML algorithms, including RF and SGTB (del Rio et al. 2014). Actually, the problem mainly lies in the *absolute rarity* of the minority class training examples (He and Garcia 2009, Weiss 2004). For example, *pressure,* the minority class for the safety outcome *energy type*, featured only 47 training examples. This is definitely not a lot of observations in absolute terms, and represents an imbalance of 1 to 60 compared to the majority class, *motion* (2,780 observations). Other categories, such as *mechanical* (74) or *biological* (108) were also severely underrepresented. For the safety outcome *body part*, the minority class (*neck*) comprised only 61 observations, as compared to the 1,532 training cases of *upper extremities* (imbalance of 1:25).

Often in such situations, the final ML models do well for the majority classes, but neglect the minorities (Sun et al. 2007, Chawla 2005, Akbani et al. 2004). This was a critical issue in this study because accurately predicting the rare categories was at least as important as predicting the majority ones.

To address class imbalance for the RF models, we used stratified oversampling (del Rio et al. 2014, Chen et al. 2004, Chawla 2002). By growing each tree of the forest on a random sample containing more training examples from the minority classes than what would have been obtained by pure chance, oversampling allowed the underrepresented concepts to become more important from the perspective of the learning algorithm, while preserving all the information from the majority categories. This strategy was implemented in R using the *sampsize* argument of the "randomForest" function (Liaw and Wiener 2002). For the SGTB models, oversampling was used ahead of model building so that the number of cases from each class matched optimal proportions. This technique produced the same effect as stratified oversampling, by rebalancing the probabilities of randomly drawing examples from each class.

One should note that improvement for the underrepresented categories is always attained at the expense of a decrease in accuracy for the majority classes, regardless of the method used to address class imbalance (Chen et al. 2004). Under the severe class imbalance we faced, attaining low error for all categories was impossible. Rather, our goal was to rebalance the overall error between all categories to improve accuracy for the minority classes without losing much accuracy for the majority categories. To achieve best performance, resampling proportions were therefore integrated to the parameter tuning protocols of RF and SGTB, following the recommendation from Sun et al. (2007). We describe these procedures in what follows.

3.4. Parameter optimization

This section describes how the optimal parameter values of the models were found. As was previously explained, one RF and one SGTB model were created for each of the four safety outcomes that were to be predicted, that is, (1) *energy type* involved, (2) *injury type*, (3) *body part* affected, and (4) *injury severity*. This gave four RF and four SGTB models. Parameter optimization is a fundamental step of statistical learning that seeks to find the optimal level of model complexity, that is, the right tradeoff between training and predictive performance (Bergstra and Bengio 2012). The overall strategy consists in searching through the parameter space and recording predictive error in terms of an objective function selected by the user. The combination of parameters minimizing the objective function gives the optimal model. The choice of the objective function and of the searching scheme is often dictated by the dimensionality of the parameter space, the computational resources available, and the nature of the ML algorithm (Claesen and De Moor 2015). In what follows, we describe the approach we adopted to tackle parameter optimization.

3.4.1. Parameter optimization for Random Forest (RF)

As already explained, the tuning parameters of RF are the total number of trees $ntree$, and the number $mtry$ of predictors randomly tested at each split. As was also already explained, class imbalance was addressed using stratified oversampling. The first step of the optimization procedure involved finding the best stratified bootstrap proportions ($sampsize$ parameter). Then, $mtry$ and $ntree$ were optimized in sequence, as shown in Figure 3.

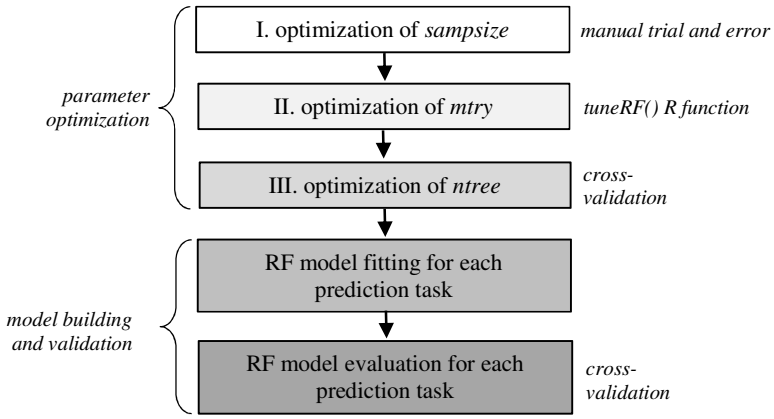

Figure 3. Overview of the parameter tuning and model evaluation procedure for RF

3.4.1.1. Step 1: Optimization of the sampsize parameter

Figure 3 shows the procedure followed to determine the best stratified oversampling proportions. Initially, each category was assigned a weight inversely proportional to the number of observations it contained. For instance, as summarized in Table 3, the safety outcome *body part* featured 5 levels: *neck* (61 training examples available), *head* (899), *trunk* (354), *upper extremities* (1532), and *lower extremities* (710). The initial weights for this safety outcome were therefore $1532/61$ for *neck*, $1532/899$ for *head,* $1532/354$ for *trunk*, $1532/1532$ for *upper extremities*, and $1532/710$ for *lower extremities*.

Randomly drawing with replacement from each class according to these weights generated samples of the original training set where each class was equally represented. Continuing with the body part example, $1{,}532$ observations were randomly sampled from each category, making for an initial balanced sample of $7{,}660$ observations.

Finally, based on the "out-of-bag" (OOB, Breiman 1996) error estimate of the resulting RF model, the classes associated with higher error rates were given more weight, and vice versa. As shown in Figure 4, this manual trial and error process was repeated until the error was evenly distributed between all classes. We used the OOB error rate estimate as a surrogate for predictive accuracy since it has been proven to be unbiased and at least as accurate as cross-validation (Wolpert and Macready 1999, Breiman 1996). Consequently, costly cross-validation procedures could be avoided at this time. Also, because testing many different combinations of weights was usually required before reaching a satisfying between-class error balance, the RF models were at this stage fitted with standard, affordable values of the *mtry* and *ntree* parameters (respectively, 20 and 81). The final weights and *sampsize* values for each model (each prediction task) are given in Table 4.