

LEVERAGING UNSTRUCTURED CONSTRUCTION INJURY REPORTS TO  
PREDICT SAFETY OUTCOMES AND MODEL SAFETY RISK USING  
NATURAL LANGUAGE PROCESSING, MACHINE LEARNING, AND  
PROBABILITY THEORY

by

ANTOINE JEAN-PIERRE TIXIER

M.S., Mechanical & Electrical Engineering, ESTP, 2011

M.S., Construction Engineering & Management, University of Colorado, 2013

A dissertation submitted to the  
Faculty of the Graduate School of the  
University of Colorado in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy

Department of Civil, Environmental, and Architectural Engineering

2015

This dissertation untitled:

"Leveraging Unstructured Construction Injury Reports to Predict Safety Outcomes and Model Safety Risk using Natural Language Processing, Machine Learning, and Probability Theory"  
has been approved by the Department of Civil, Environmental, and Architectural Engineering

---

Professor Matthew R. Hallowell

---

Professor Balaji Rajagopalan

---

Professor Paul M. Goodrum

---

Assistant Professor Daniel Tran

---

Assistant Professor Joseph R. Kasprzyk

Date \_\_\_\_\_

The final copy of this dissertation has been examined by the signatories, and we  
Find that both the content and the form meet acceptable presentation standards  
Of scholarly work in the above mentioned discipline.

Antoine Jean-Pierre Tixier (PhD, Civil Engineering).

Leveraging Unstructured Construction Injury Reports to Predict Safety Outcomes and Model Safety Risk using Natural Language Processing, Machine Learning, and Probability Theory.

Dissertation directed by Professor Matthew Ryan Hallowell.

Construction is one of the most dangerous industries in the United States and throughout the globe. Despite the abundant research that has been motivated by the very high socio-economic costs induced by accidents, safety performance plateaus and injuries still occur at an unacceptable, disproportionate rate. The paradox is that at the same time, huge databases of valuable textual injury reports are left mostly unused, because of the lack of a conceptual framework to readily extract usable knowledge from them and because manual content analysis is very expensive.

Not only do these vast amounts of candid narratives represent a wealth of valuable lessons to be learned, but they could also transform the way safety is approached in construction. From mostly being dealt with through the analysis of subjective, aggregated, or secondary data; expert-opinion; and according to a strictly regulatory and managerial perspective; construction safety could become an empirical, data-driven science, where objective, quantitative techniques such as Machine Learning and statistical modeling could play a determinant role.

To provide a proof for this concept, we (1) developed a Natural Language Processing tool to automatically extract fundamental attributes and outcomes from unstructured textual injury reports and remove the needs for manual content analysis; (2) explored the interplay and detected clashes between attributes using unsupervised clustering and network analysis techniques; (3) applied supervised Machine Learning algorithms to capture the mapping between attribute and outcome data and predict various safety outcomes; and (4) proposed a new way to model and simulate construction safety risk using probability theory tools such as Kernel Density Estimates and Copulas.

At every level, results are promising and show that by following the aforementioned pipeline, it is possible to better understand and predict injuries, simply from raw textual data. We hope this research shows that adopting a data-driven approach could lead to better-informed, safer decision-making, and improve safety performance in construction.

*To my father Daniel,  
chess master, expert sailor, architect, brilliant inventor, beloved professor, bridge champion.*

*Now the years are rolling by me  
They are rocking evenly  
I am older than I once was  
Younger than I'll be  
But that's not unusual  
No, it isn't strange  
After changes upon changes  
We are more or less the same  
After changes, we are more or less the same*

## **ACKNOWLEDGMENTS**

First and foremost, I am grateful to my advisor Professor Matthew Hallowell for all he has done for me. In the fall of 2011, Dr. Hallowell offered me a position in his research group even though I was a Master student recently arrived from France with no research experience and limited English skills. I want to thank him for trusting me, giving me a chance, and for all his patience and forgiveness. After I completed my Master, he provided me with a very exciting and interesting PhD opportunity. Looking back, this PhD was a life-changing experience, since it made me discover my passion for Machine Learning, which I now follow.

Prof. Hallowell was a great advisor. He taught me everything I know about research. He is intelligent, curious, open-minded, and we share the same strong interest for scientific exploration, innovation, and thinking outside the box. The exploratory and interdisciplinary nature of our work turned out extremely instructive and highly enjoyable. My best graduate school memories are the brainstorming sessions in his office.

Dr. Hallowell is also an affable, enthusiastic and caring individual, very generous with his time. At every step of my graduate studies, he was always very responsive when I needed help. He was also a great supporter who provided me with much appreciated research and teaching assistantships and believed in my abilities and my work sometimes more than I did myself. He always found the right words to reignite motivation. Without his continuous support and inspiration, I would have never made it through.

Finally, Prof. Hallowell had a clear overall research vision for the projects I was involved in, which made my work so much easier because I knew where I had to go. At the same time, he was flexible enough to allow me to choose my own paths and select (or invent) my own tools to get there. This unique blend of leadership and room for creativity and autonomy made my PhD an unforgettable experience.

Prof. Hallowell will be a role model for me as I continue in life, both from a professional and personal standpoint. I feel privileged to have been advised by him, and I hope the future will bring many opportunities for fruitful collaboration and partnership.

Sincere thanks also go to Professor Balaji Rajagopalan for his central role during my doctoral journey. His great ideas and advice, willingness to share knowledge and material, kindness, positive attitude and continuous support and availability were appreciated beyond measure. Furthermore, I want to thank him for introducing me to the wonderful R programming language and to the art and science of statistical modeling. CVEN 6833 and CVEN 5454 are two of the most interesting and challenging courses I have ever taken. I will long remember my brief excursion into the world of atmospheric rivers and decadal oscillations as a most exciting one. I hold Dr. Rajagopalan in high esteem and he will remain a source of inspiration for the rest of my career.

I am also indebted to my other committee members, Professors Paul Goodrum, Daniel Tran and Joseph Kasprzyk, in particular for their insightful questions and suggestions during my comprehensive examination. Their feedback has undoubtedly improved the quality of my work.

My colleagues from the Colorado Construction Safety Laboratory Marc Prades Villanova and Matthieu Desvignes should also be acknowledged for their rigorous management of the manual content analysis teams over the first two semesters of my doctorate. The members of their teams naturally deserve mention: Alex Aguilar, Alex Albert, Dillon Alexander, Siddarth Bandari, Daniel Hansen, Spencer Lacy, and Zach Noonan. Thank you all for the good work. You ensured the successful completion of the initial data collection and crucial feature engineering phases, as well as the proper calibration of the NLP tool, which were critical to all downstream analyses.

I would also like to thank my classmates, labmates and friends Alex Albert, Rayyan Alsamadani and Marc Prades, with whom I enjoyed numerous lively discussions about work and life; my officemates Maryam Sanaei, Mohammed Albattah, and Omar Alruwaythi, for always maintaining a pleasant, cooperative and productive atmosphere; Pascal and Sukmi Ledru for their hospitality and continuous friendship and kindness during my time in Colorado; my classmate Srijita Jana for sharing helpful homework sessions and introducing me to the excellent Knitr library of Yihui Xie, which proved extremely useful over the course of my PhD, and finally Louis Aslett for

his invaluable R Studio Amazon EC2 IDEs, without which I would not have been able to get results so fast and so easily.

The international student that I was deeply appreciated the warm, family-like atmosphere created by the young and dynamic faculty members of the Construction Engineering and Management program: Matthew Hallowell, Amy Javernick-Will, Paul Goodrum, Keith Moolenaar, and Matthew Morris. Thank you for the end-of-semester parties, annual research retreats in Breckenridge, and basketball sessions.

Last but not least, I am thankful to Pamela Williamson, our exceptional graduate program coordinator, Maria Zellar Maxim from the International Student Services, and Kendra Zafitaros from the International Tax Services. Their availability, competence, guidance, and timely reminders helped me so much meeting the numerous administrative deadlines and requirements. I would have felt lost without them.

## TABLE OF CONTENTS

ABSTRACT .....	i
ACKNOWLEDGMENTS .....	v
LIST OF FIGURES .....	xiii
LIST OF TABLES .....	xvi
CHAPTER 1: INTRODUCTION.....	1
OBSERVED PROBLEM .....	2
RESEARCH OBJECTIVES.....	3
DISSERTATION FORMAT .....	4
Chapters 1 and 6 .....	4
Chapter 2 .....	5
Chapter 3 .....	5
Chapter 4 .....	5
Chapter 5 .....	6
PUBLICATIONS .....	6
CHAPTER 2: AUTOMATED CONTENT ANALYSIS FOR CONSTRUCTION SAFETY: A NATURAL LANGUAGE PROCESSING SYSTEM TO EXTRACT PRECURSORS AND OUTCOMES FROM UNSTRUCTURED INJURY REPORTS .....	8
ABSTRACT .....	9
MOTIVATION.....	9
BACKGROUND: ATTRIBUTE-BASED APPROACH TO CONSTRUCTION SAFETY.....	11
BACKGROUND: NATURAL LANGUAGE PROCESSING .....	15

Selection of an appropriate natural language processing method .....	16
Design of a rule-based natural language processing system .....	18
Validation of the system and measurement of reliability .....	34
<b>CONCLUSION, LIMITATIONS, AND RECOMMENDATIONS .....</b>	<b>38</b>
Limitations.....	39
Recommendations for future research.....	40
<b>CHAPTER 3: SAFETY CLASH DETECTION FOR BIM, WORK PACKAGING, AND OPERATIONS:</b>	
<b>IDENTIFYING SAFETY INCOMPATIBILITIES AMONG FUNDAMENTAL CONSTRUCTION</b>	
<b>ATTRIBUTES BY APPLYING GRAPH MINING AND HIERARCHICAL CLUSTERING TO ATTRIBUTE</b>	
<b>DATA SETS.....</b>	<b>42</b>
<b>ABSTRACT .....</b>	<b>43</b>
<b>INTRODUCTION AND MOTIVATION.....</b>	<b>43</b>
<b>BACKGROUND AND POINT OF DEPARTURE .....</b>	<b>45</b>
Construction safety risk analysis .....	45
Modeling and managing safety in BIM.....	47
Point of departure .....	48
<b>ANALYSIS .....</b>	<b>50</b>
Presentation of the data set.....	50
Overview of the methods employed.....	51
Graph Mining .....	53
Graph mining: Results .....	59
Hierarchical Clustering on Principal Components (HCPC) .....	72

Hierarchical clustering on principal components: results.....	77
CONCLUSION .....	82
<b>CHAPTER 4: APPLICATION OF MACHINE LEARNING TO CONSTRUCTION INJURY PREDICTION....</b>	<b>84</b>
ABSTRACT .....	85
INTRODUCTION AND MOTIVATION.....	85
BACKGROUND AND POINT OF DEPARTURE.....	88
Why does safety outcome prediction matter?.....	88
Limitations of previous work on attribute-based construction safety .....	89
Previous use of machine learning (ML) in construction .....	92
Goal of this study.....	94
Characteristics of the data set .....	95
<b>APPLICATION OF MACHINE LEARNING (ML)</b> .....	<b>98</b>
Ensemble learning .....	100
Classification and regression trees (CART) .....	101
Bagging .....	107
Random Forest (RF) .....	109
Boosting.....	111
Class imbalance issue .....	117
Parameter optimization.....	120
Measuring predictive skill with RPSS .....	130
<b>RESULTS AND INTERPRETATION .....</b>	<b>133</b>

Predictive skill .....	133
Variable importance .....	137
<b>CONCLUSIONS, LIMITATIONS AND RECOMMENDATIONS .....</b>	<b>142</b>
Contributions to theory .....	142
Contributions to practice .....	144
<b>CHAPTER 5: CONSTRUCTION SAFETY RISK MODELING AND SIMULATION .....</b>	<b>146</b>
<b>ABSTRACT .....</b>	<b>147</b>
<b>INTRODUCTION AND MOTIVATION.....</b>	<b>147</b>
<b>BACKGROUND AND POINT OF DEPARTURE.....</b>	<b>150</b>
Data .....	151
Level of analysis.....	152
<b>UNIVARIATE ANALYSIS.....</b>	<b>153</b>
Attribute-level safety risk .....	153
Report-level safety risk.....	157
The probability distribution of construction safety risk resembles that of many natural phenomena .....	158
Why does construction safety risk follow a power law? .....	164
Univariate modeling .....	166
Limitations of traditional parametric techniques.....	167
Univariate construction safety risk generator .....	168
<b>BIVARIATE ANALYSIS.....</b>	<b>171</b>
Bivariate modeling .....	173

Bivariate construction safety risk generator .....	175
Computing risk escalation potential based on simulated values .....	180
LIMITATIONS .....	181
CONCLUSIONS AND RECOMMENDATIONS .....	182
AKNOWLEDGMENTS.....	183
CHAPTER 6: CONCLUSIONS AND NEXT STEPS .....	184
REFERENCES.....	188

# LIST OF FIGURES

## CHAPTER 2

Figure 1. Overarching tool process flow .....	19
Figure 2. Tool building and validation process .....	20
Figure 3. Cleaned and structured incident report .....	30
Figure 4. Detection of the variable <i>slippery walking surface</i> based on a single complex.s() statement .....	30
Figure 5. Faulty detection of the variable <i>slippery walking surface</i> by a single complex.s() statement .....	30
Figure 6. Two complex.s() statements allow the false alarm to be avoided.....	30
Figure 7. Scanning of injury reports, and post-processing .....	32

## CHAPTER 3

Figure 1. Structure of the attribute data set used in this study.....	51
Figure 2. Top nodes for (a) eigenvector centrality, (b) closeness, and (c) top edge for edge betweenness with two natural communities .....	57
Figure 3: Overall graph mining procedure .....	59
Figure 4: attribute co-occurrence graph for <i>struck by or against</i> (2,389 reports) .....	62
Figure 5: attribute co-occurrence graph for <i>fall on same or to lower level</i> (350 reports) .....	64
Figure 6: attribute co-occurrence graph for <i>overexertion</i> (570 reports) .....	66
Figure 7: attribute co-occurrence graph for <i>caught-in or compressed</i> (567 reports).....	67
Figure 8: attribute co-occurrence graph for <i>exposure to harmful substance</i> (525 reports) .....	70
Figure 9: Hierarchical Clustering on Principal Components (HCPC) steps.....	72
Figure 10. Agglomerative hierarchical clustering illustration for r = 11 observations.....	75

## CHAPTER 4

Figure 1. The derivation of predictive models from injury reports is enabled by the attribute-based framework ...	87
Figure 2. Limitations of past research and solutions brought by the present study.....	91
Figure 3. Practical use of the predictive models built in this study .....	94

Figure 4: ML versus traditional statistics (adapted from Breiman 2001a).....	99
Figure 5. The two methods used in this study, RF and Boosting, are both ensemble learning techniques. Besides that, they widely differ. .....	101
Figure 6: CART example in the two-dimensional case.....	105
Figure 7: From CART to Random Forest.....	109
Figure 8. Schematic representation of a small Random Forest ( <i>ntree</i> = 8).....	110
Figure 9: from Boosting to Stochastic Gradient Boosting .....	113
Figure 10. Schematic representation of Least Squares Gradient Boosting with <i>n.tree</i> = 5.....	116
Figure 11. Overview of the parameter tuning and model evaluation procedure for RF.....	122
Figure 12. Optimization of the <i>sampsize</i> parameter for RF.....	123
Figure 13. “Leave-5%-out” cross-validation procedure .....	126
Figure 14. Parameter optimization and model evaluation procedure used for SGTB .....	129
Figure 15. Predictive skill for the first three prediction tasks, as measured by RPSS recorded in 36 runs of cross-validation .....	135
Figure 16. Predictive skill of the models for the last prediction task, as measured by RPSS recorded in 36 runs of cross-validation .....	137
Figure 17. Variable importance for <i>body part</i> .....	138
Figure 18. Variable importance for <i>injury type</i> .....	139
Figure 19. Variable importance for <i>energy type</i> .....	141
Figure 20. Variable importance for <i>injury severity</i> .....	141
<b>CHAPTER 5</b>	
Figure 1. Overarching research process: from raw injury reports to safety risk analysis.....	150
Figure 2. Histogram of original observations (n=814) with boundary corrected KDE of the simulated observations (n=10 <sup>5</sup> ) .....	162
Figure 3. Comparison of safety risk with natural phenomena.....	162
Figure 4. Bivariate construction safety risk.....	172

Figure 5. Bivariate histogram of construction safety risk (25 by 25 grid).....	172
Figure 6. Nonparametric Copula density estimate with original pseudo observations .....	178
Figure 7. Simulated risk values in rank space (n=10 <sup>5</sup> ) .....	178
Figure 8. Bivariate construction safety risk.....	179
Figure 9. Simulated risk values in original space (n=10 <sup>5</sup> ) .....	179

## LIST OF TABLES

### CHAPTER 2

Table 1. Context-free validated injury precursors from Desvignes (2014) .....	13
Table 2. Outcome categories from Prades Villanova (2014) and Desvignes (2014) .....	14
Table 3. Examples of injury reports from Desvignes et al. (2014), with attributes and outcomes .....	18
Table 4. Comprehensive list of functions used for writing rules.....	27
Table 5. Example of the system's output for 12 injury reports .....	33
Table 6. Retrieval matrix (adapted from Buckland and Grey 1994). ....	34
Table 7. System's performance at each step of the tuning process .....	37

### CHAPTER 3

Table 1. Attribute counts in our data set.....	52
Table 2. Top elements for eigenvector centrality, closeness, and edge betweenness.....	71

### CHAPTER 4

Table 1. Eighty context-free validated injury precursors from Tixier et al. (2016a).....	96
Table 2. Safety outcomes predicted.....	96
Table 3. Number of observations for each level of the four safety outcomes predicted .....	97
Table 4. Optimal weights and values of the <i>sampsize</i> parameter for each prediction task (RF) .....	124
Table 5. Optimal parameter values for each prediction task (RF).....	127
Table 6. Optimal resampling proportions and final numbers of cases for each prediction task (SGTB) .....	129
Table 7. Optimal parameter values for each prediction task (Boosting) .....	129
Table 8. Calculation of <i>RPSforecast</i> .....	132
Table 9. Calculation of <i>RPSreference</i> .....	132
Table 10. Example of probabilistic forecasts issued by the SGTB model for <i>injury type</i> .....	135
Table 11. Mean and median RPSS for each prediction task.....	137

## CHAPTER 5

Table 1. Relative risks and counts of the P = 77 injury precursors.....	151
Table 2. Counts of injury severity levels accounted for by each precursor.....	154
Table 3. Severity level impact scores .....	154
Table 4. Quantile estimates for the risk based on real outcomes.....	169
Table 5. Proposed ranges for the risk based on real outcomes .....	170
Table 6. Quantile estimates for the risk based on worst potential outcomes.....	177
Table 7. Proposed ranges for the risk based on worst potential outcomes .....	177

## **CHAPTER 1: INTRODUCTION**

## OBSERVED PROBLEM

Construction is constantly ranked as one of the most dangerous industries worldwide. In the United States, despite the improvements that followed the Occupational Health and Safety Act of 1970, the construction industry still accounts for 17% of all work-related deaths (CPWR 2013), while only employing 7% of the national workforce (BLS 2011). 781 workers died on the job in 2011 (BLS 2014), and in total, 21,000 fatalities have been reported between 1992 and 2010 (CPWR 2013), which approximately equates 1,000 casualties per year. Construction also suffers a disproportionate share of occupational diseases. For instance, 16% of all work-related elevated blood lead levels cases occurred in construction (CPWR 2013). The total cost of construction injuries in the United States has been estimated to approach \$15 billion every year (BLS 2011).

As a response to this alarmingly poor performance, construction safety research abounds. However, most safety analyses rely on subjective, secondary, or aggregated data (Prades Villanova 2014). The paradox is that major construction firms and federal agencies have put together huge databases of digital injury-related events and near misses in the past few decades, which represents a wealth of empirical data. Unfortunately, the greater part of this valuable knowledge has been left unstructured and unexploited so far, because of the lack of an adapted methodology, and because manual content analysis is fraught with time, labor, and organizational limitations. Finally, the major part of previous construction safety efforts have been limited to the study of specific trades, tasks, and activities (Prades Villanova 2014).

To summarize, previous work (1) is not empirically grounded, (2) does not translate well to other work scenarios, and (3) overlooks the global, multifactorial nature of safety and the interactions inherent to complex and dynamic environments such as construction sites.

The attribute-based framework introduced by Esmaeili and Hallowell (2012) offers a way to jointly overcome the three aforementioned problems. First, it was designed to be primarily applied to injury reports, a major, almost unlimited source of objective raw data. Second and third, drawing from genetics, it shows that a *finite* set of binary features of the work can be used to encode the signature of a potentially *unlimited* number of construction

situation in a universal and standard way, regardless of the task, trade, or industry sector. These binary features, also called *fundamental construction attributes*, pertain to construction means and methods, environmental conditions, and human behavior. Since they are observable before accident occurrence, attributes are also called *injury precursors*. The two terms are used interchangeably in this dissertation.

## RESEARCH OBJECTIVES

While making great strides, Esmaeili and Hallowell (2012) did not offer a very mature and comprehensive list of attributes, nor an automated way of automatically extracting these attributes from unstructured text. The high costs of manual content analysis remained. Two recent studies (Prades Villanova 2014, Desvignes 2014) addressed the first limitation by widening and refining Esmaeili and Hallowell (2012)'s framework. The present research capitalized on this more mature, robust version of the framework to address the following overall research objectives:

*This research seeks to improve safety performance in the construction industry by learning from large amounts of unstructured textual injury reports.*

More specifically, this main objective was broken down into the following four sub-objectives:

**Objective 1:** develop a Natural Language Processing tool to remove the needs for manual content analysis of injury reports in order to extract attribute and outcomes data from large numbers of injury reports and unlock the full potential of the attribute framework refined by Prades Villanova (2014) and Desvignes (2014).

**Objective 2:** propose an approach based on unsupervised Machine Learning to find interesting patterns of variability in the attribute data and highlight safety critical combinations of attributes.

**Objective 3:** propose an approach based on supervised Machine Learning to encode the mapping between precursors and outcomes, in order to predict and better understand injury occurrence.

**Objective 4:** extend safety risk from the attribute to the observational level. Use probability theory to model and simulate safety risk.

In order to meet these goals, the research had to be divided into four complementary phases. Each phase is discussed in details in the core chapters of this dissertation. They are briefly presented next.

## **DISSERTATION FORMAT**

The core of this dissertation is divided into four, independent manuscripts (Chapters 2 to 5). Each section is an extended version of an article submitted to a peer-reviewed academic journal (see subsection “List of Publications”). However, note that to minimize redundancy and improve clarity, all references are provided in a single list at the end of this document.

Chapters 2 and 4 have already been published, Chapters 3 and 5 are still under review. Because each chapter uses the same underlying conceptual framework and parts of the same data sets, they exhibit some minor overlap, notably in their respective introduction and literature review sections. However, care was taken to avoid repetition as much as possible. In what follows, the gist of each chapter is provided.

## **Chapters 1 and 6**

These two chapters respectively serve as the overall introduction and conclusion of this dissertation.

## **Chapter 2**

The second chapter introduces the rule-based Natural Language Processing (NLP) tool that was developed to extract fundamental attributes and safety outcomes from unstructured textual injury reports. All subsequent core chapters use data extracted by this tool.

## **Chapter 3**

The third manuscript applies unsupervised Machine Learning techniques, notably hierarchical clustering and community detection in graphs, to attribute data. In particular, it focuses on the detection of safety incompatibilities among attributes, what we define as “safety clashes”. The proposed representation of the data as weighted undirected graphs and the genetics-inspired theory that construction accidents may be explained by perturbations in attribute-attribute networks are also valuable contributions. The attribute-based nature of the methodology makes it ideal for integration in BIM and work packaging software, which would enable safety clashes learned from data to be automatically detected in early stages of a project.

## **Chapter 4**

This chapter uses cutting-edge supervised Machine Learning algorithms to encode the mapping between attributes and safety outcomes and predict the occurrence of construction accidents. The high skill reached suggests that construction injuries do not occur in a chaotic fashion but that rather they may be predictable from features of the work that are easily observable ahead of accident occurrence. It also validates the attribute-based framework and NLP tool presented in Chapter 2. The proposed models can be used by practitioners at every level of the construction process to proactively implement adequate remediation strategies and prevent injuries.

## **Chapter 5**

The fifth chapter formally defines univariate and bivariate construction safety risk at the situational level (report level here). It provides simple yet powerful tools drawn from the state-of-the-art in hydroclimatology and insurance statistics to model and simulate both types of risk in a fully nonparametric and data-driven way. The proposed methodology can be used in practice to better estimate and provision for risk.

## **PUBLICATIONS**

Chapters 2 to 5 are extended versions of articles that were submitted to peer-reviewed scholarly journals. Some have already been published, while some are still in the review process. For clarity, and for the convenience of the reader, we provide the corresponding references below.

### **Chapter 2:**

Tixier, A. J. P., Hallowell, M. R., Rajagopalan, B., & Bowman, D. (2016a). Automated content analysis for construction safety: A natural language processing system to extract precursors and outcomes from unstructured injury reports. *Automation in Construction*, 62, 45-56.

### **Chapter 3:**

Tixier, A. J. P., Hallowell, M. R., Rajagopalan, B., & Bowman, D.. Construction Safety Clash Detection: Identifying Safety Incompatibilities among Fundamental Attributes using Data Mining. Submitted to *Automation in Construction*.

### **Chapter 4:**

Tixier, A. J. P., Hallowell, M. R., Rajagopalan, B., & Bowman, D. (2016b). Application of machine learning to construction injury prediction. *Automation in Construction*, 69, 102-114.

**Chapter 5:**

Tixier, A. J. P., Hallowell, M. R., and Rajagopalan, B.. Construction Safety Risk Modeling and Simulation.

Submitted to *Risk Analysis*.

**CHAPTER 2: AUTOMATED CONTENT ANALYSIS FOR CONSTRUCTION  
SAFETY: A NATURAL LANGUAGE PROCESSING SYSTEM TO EXTRACT  
PRECURSORS AND OUTCOMES FROM UNSTRUCTURED INJURY  
REPORTS**

## **ABSTRACT**

In the United States like in many other countries over the world, construction workers are more likely to get injured on the job than workers in any other industry. This poor safety performance is responsible for huge human and financial losses and has motivated extensive research. Unfortunately, safety improvement in construction has decelerated in the last decade and traditional safety programs have reached saturation. Yet, major construction companies and federal agencies possess a wealth of empirical knowledge in the form of huge databases of digital construction injury reports. This knowledge could be used to better understand, predict, and prevent the occurrence of construction accidents. Unfortunately, due to the lack of a clear methodology and the high costs of manual large-scale content analysis, these valuable data have still to be extracted and leveraged. Recent research has proposed a framework allowing meaningful empirical data to be extracted from accident reports. However, the resource limitations inherent to manual content analysis still remain. The present study tested the proposition that manual content analysis of injury reports can be eliminated using natural language processing (NLP). This paper describes (1) the overall strategy and methodology used in developing the system, and specifically how key challenges with decoding unstructured textual reports were overcome; (2) how the system was built through an iterative process of coding and testing against results from a team of seven independent analysts; and (3) the implications and potential uses of the data extracted. The results indicate that the NLP system is capable of quickly and automatically scanning unstructured injury reports for 101 attributes and outcomes with over 95% accuracy. The main contribution of this research is to empower any organization to quickly obtain a large and highly reliable structured attribute and outcome data set from their databases of unstructured accident reports. Such structured data are a necessary prerequisite to the application of statistical modeling techniques allowing the extraction of new safety knowledge and finally the amelioration of safety management.

## **MOTIVATION**

Construction is constantly ranked as one of the most dangerous industries worldwide. In the United States, despite the improvements that followed the Occupational Health and Safety Act of 1970, construction still accounts for

17% of all work-related deaths while only employing 7% of the national workforce (CPWR 2013). In fact, according to the Bureau of Labor Statistics (2015), more than 800 construction workers die on the job each year. What is even more alarming is that the colossal human and financial costs induced by fatalities and injuries are expected to escalate with the 33% construction employment growth projections in the 2010-2020 decade, which is more than twice the overall anticipated economic growth (CPWR 2013).

Despite the abundant research that has been motivated by the aforementioned alarming injury and fatality rates, safety performance in construction has been plateauing in recent years and the implementation of effective injury prevention practices has reached saturation (Esmaeili and Hallowell 2011a). Fortunately, risk-based approaches are emerging and show promise for safety improvement through proactive decision-making. For example, Baradan and Usman (2006) compared the risk of building trades, Hallowell and Gambatese (2009a) quantified the safety risk for various activities required to construct concrete formwork, and Shapira and Lyachin (2009) studied the impact of tower cranes on jobsite safety. However, these approaches are currently limited because (1) they focus on specific activities and trades without considering the temporal and spatial interactions among risk factors; (2) they are not based on empirical data; and (3) they are limited in scope of application (Prades Villanova 2014, Sacks et al. 2009). Consequently, existing models do not translate well to other work scenarios, and do not capture the dynamics of construction work, where trades and activities constantly interact (Sacks et al. 2009, Helander 1991). To overcome these limitations, Esmaeili and Hallowell (2012, 2011b) proposed a unified attribute-based framework that allows risk factor and outcome variables to be extracted from accident reports. Although this method shows promise, it requires the analysis of large numbers of reports, which is laborious and resource-intensive when performed manually (Prades Villanova 2014, Desvignes 2014).

Motivated by high injury rates and inspired by the attribute-based approach to construction safety, we tested the proposition that attribute and outcome data can be automatically and accurately extracted from unstructured injury reports using natural language processing (NLP).

## **BACKGROUND: ATTRIBUTE-BASED APPROACH TO CONSTRUCTION SAFETY**

The attribute-based approach to construction safety theorizes that any construction situation can be uniquely and comprehensively characterized by a finite number of observable fundamental construction site attributes (Prades Villanova 2014, Desvignes 2014). These basic elements are context-free, universal, and pertain to construction means and methods, environmental conditions, and human factors. For instance, in the following excerpt of an injury report: “employee was welding overhead and wind shifted, resulting in discomfort to left eye”, three fundamental attributes can be identified: (1) *welding*, (2) *working overhead*, and (3) *wind*.

Although this approach is simple, it is powerful. First, from this perspective, any incident can be viewed as the resulting outcome of the presence of a worker and the joint presence of some fundamental attributes. This is why attributes are also called *injury precursors*, or simply *precursors*. In what follows, we will use these terms interchangeably. It is important to note that precursors should be observable before an injury occurs. *Falling object*, for example, is not a precursor, it is an outcome. On the other hand, *object at height* is a precursor. Second, the attribute-based approach shows that incident reports contain valuable information. As illustrated in the previous example, descriptors of the work environment and outcomes can be extracted even from brief reports. Finally, this information is authentic since it is simply based on objective narratives of discrete events.

A connection with genetics can naturally be made with this style of analysis: every person is unique, but their genetic information is entirely encoded by combinations of a finite number of basic universal building blocks that constitute their DNA. The attribute-based approach to construction safety is built upon a similar theory that by identifying fundamental and universal construction injury precursors, understanding how they interact, and modeling how they shape risk and create unsafe work conditions, it may be possible to better understand the true nature of, predict, and prevent the occurrence of construction injuries. Historically, scientific understanding of complex phenomena has always improved when breaking down convoluted systems into fundamental constituents that individually are easier to comprehend. A fascinating recent example is the Human Genome Project (Collins et al. 2003), which allowed sequencing and mapping of about 30,000 genes, unlocking the

structure of human DNA. Similarly, the finite element method, a numerical technique used in many quantitative disciplines of engineering, is built on the theory that complicated continuous structures and objects can be represented by a finite number of geometrically simpler pieces (Zienkiewics 1971).

Esmaeili and Hallowell (2012, 2011b) conducted the first attribute-level risk analyses in construction by analyzing 150 fall and 300 struck-by injury cases (respectively) from national databases. Through this analysis, they reported 14 and 34 fundamental attributes. More recently, a team of eight researchers performed a manual content analysis of 2,201 industrial, energy, infrastructure, and mining injury reports gathered from 476 contractors, allowing the initial lists to be refined and broadened to 80 precursors (Prades Villanova 2014, Desvignes 2014). These precursors are summarized in Table 1.

The validity of the content analysis and relevance of the attributes was ensured 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. In these studies, attributes were classified in three categories: upstream, transitional, and downstream. 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.

**Table 1. Context-free validated injury precursors from Desvignes (2014)**

<b>UPSTREAM</b>			
Cable tray	Rebar	Screw	
Cable	Scaffold	Slag	
Chipping	Soffit	Spark	
Concrete liquid	Spool	Slippery walking surface	
Concrete	Stairs	Small particle	
Conduit	Steel sections	Adverse low temperatures	
Confined workspace	Stripping	Unpowered tool	
Congested workspace	Tank	Unstable support/surface	
Crane	Unpowered transporter	Wind	
Door	Valve	Wrench	
Dunnage	Welding	Lifting/pulling/manual handling	
Electricity	Wire	Light vehicle	
Formwork	Working at height	Exiting/transitions	
Grinding	Working below elevated workspace/material	Sharp edge	
Grout	Drill	Splinter/sliver	
<b>TRANSITIONAL</b>			
Guardrail/handrail	Bolt	Repetitive motion	
Heat source	Cleaning	Working overhead	
Heavy material/tool	Forklift		<b>DOWNSTREAM</b>
Heavy vehicle	Hammer	Improper body position	
Job trailer	Hand size pieces	Improper procedure/inattention	
Lumber	Hazardous substance	Improper security of materials	
Machinery	Hose	Improper security of tools	
Manlift	Insect	No/improper PPE	
Stud	Ladder	Object on the floor	
Object at height	Mud	Poor housekeeping	
Piping	Nail	Poor visibility	
Pontoon	Powered tool	Uneven walking surface	

In addition to the 80 precursors presented in Table 1, Prades Villanova (2014) and Desvignes (2014) also extracted safety outcomes from accident reports, including injury type, injury severity, body part affected, and energy type involved. The variables belonging to these categories are listed in Table 2. The injury codes, severity levels, and body divisions included in Table 2 are consistent with OSHA definitions and past research (Hallowell 2008). Types of energy were extracted based on the theory that any injury is caused by the release of some form of energy (Fleming 2009, Haddon 1973). For instance, a suspended load is a source of *gravity* and *motion*, welding releases *radiation*, and waterproofing substances, solvents, or concrete in its liquid form are sources of *chemical* energy. Additional definitions and examples can be found in Albert et al. (2014). We will sometimes jointly refer to attributes and outcomes as *variables* in what follows, for simplicity.

**Table 2. Outcome categories from Prades Villanova (2014) and Desvignes (2014)**

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

Although the work of Esmaeili and Hallowell (2012, 2011b), Prades Villanova (2014), and Desvignes (2014) made important contributions to attribute-level safety analysis, the manual content analysis procedures used were time consuming, limiting the number of reports that could be analyzed in a reasonable research effort, and thereby the emergence of trends and patterns in the data extracted. For example, Desvignes (2014) only used a random subset of 1,280 reports from a larger set of 4,458 available reports because of time and resource limitations. In addition, even when a rigorous protocol is followed, it is never possible to entirely eliminate inconsistencies among human coders. For all these reasons, resorting to manual content analysis to systematically mine large databases of construction injury reports is not viable. In order to eliminate the needs for manual analysis, and allow large databases of injury reports to be leveraged, we introduce in this study a fully automated and highly accurate NLP system.

In construction safety research, the only known attempt of automatically analyzing injury reports was made by Esmaeili (2012). In this study, high severity injury reports from national databases were scanned for 22 attributes with commercial software. Though this effort involved automated attribute extraction from injury reports for the first time, it suffers some limitations. First, the reliability of the attribute identification and keyword validation process is questionable because fewer than 500 accident reports were used to identify attributes and tune keywords, and a percent agreement score of 0.7 was set as the accuracy threshold. This is a rather low value, especially since percent agreement is known to be a lenient metric that inflates agreement in all cases (Lombard et al. 2002, Iacobucci and Dawn 2001). Second, the validated list of keywords, and explanations about the automated content analysis process were not provided, making replication of the work and assessment of the

quality of the data obtained impossible. Third, only high severity struck-by injuries were studied, which significantly narrowed the breadth of attributes that could be identified and the quantity, relevance and generalizability of the keywords found. Fourth, some precursors were defined in opposition with the conceptual attribute-based framework. For instance, *falling object*, *structure collapse*, or *falling out from heavy equipment* cannot be considered injury precursors, because they are outcomes rather than observable characteristics of the jobsite. This paper seeks to collectively address these limitations.

## **BACKGROUND: NATURAL LANGUAGE PROCESSING**

Natural Language Processing (NLP) is a very active and rapidly evolving interdisciplinary area of research that deals with the comprehension and analysis of human-produced texts by computers (Chowdhury 2003). NLP lies at the confluence of statistics, linguistics, and computer science, and aims at achieving human-like natural language understanding (Liddy 2001). Applications of NLP include speech recognition, machine translation, and automated content analysis (Manning and Schuetze 1999).

Automated content analysis is increasingly being used for a variety of applications. This can be explained by the needs to make sense of and leverage the fast-growing volume of digital information (Bai 2011). In construction, even a small project generates a lot of electronic information in the form of specifications, digital drawings, process control, inventory management, cost estimating, scheduling, and other documentation (Soibelman et al. 2008). For several years, major companies and federal agencies have also been constituting extensive databases of electronic injury and near miss reports as part of their safety programs.

In a recent study, Francis and Flynn (2010) attempted to categorize insurance claim descriptions into four categories based on keywords, and then used the clustering to predict claim severity. For instance, the claims corresponding to car accidents were automatically extracted based on the keywords *hit*, *travel* and *vehicle*. In another study, 10,000 traffic incident reports were automatically categorized into topics using Latent Dirichlet

Allocation and incorporated into predictive models to forecast time-to-clearance and improve traffic management in real time (Pereira et al. 2013). In the construction industry, text analytics have been used to classify project documents (Al Qady and Kandil 2014, Caldas and Soibelman 2003), to retrieve computer aided drawings from databases (Yu and Hsu 2013), to automate knowledge extraction from narratives and represent it as a map (Yeung et al. 2014), and to structure and manage safety knowledge in order to support Job Hazard Analysis (Chi et al. 2014).

Creating and validating a system that understands natural language is very challenging. Natural languages are complex, consisting of a catalogue of words, called a lexicon, and set of structural rules, called a grammar, that allows meaning to be built by combining words into sentences (Manning and Schuetze 1999). Historically, the most serious obstacle to effectively analyze naturally occurring language has been the difficulty in accurately modeling grammars (Hindle 1989). Additionally, many words have several meanings, making the use of word sense disambiguation indispensable.

Most modern NLP tools use machine learning (ML) algorithms and statistical modeling to overcome the aforementioned barriers. Some of the most widely used techniques in text analytics are clustering algorithms, such as K-means, and classification algorithms, such as k-nearest neighbors, naïve Bayes, and Support Vector Machines (Verma et al. 2011, Prabowo and Thelwall 2009, Karatzoglou et al. 2010, Joachims 1998). Some popular methods also include Random Forest (Grimmer and Stewart 2013), graph theory (Rousseau et al. 2015, Ferrer i Cancho and Sole 2001), Bayesian and Markov models (Bai 2011), and Latent Dirichlet Allocation (Pereira et al. 2013).

### **Selection of an appropriate natural language processing method**

In developing the automated content analysis tool, we faced a dilemma. Ideally, ML algorithms would have been applied to the available data manually-coded by Prades Villanova (2014) and Desvignes (2014). Unfortunately,

these techniques, like Support Vector Machines, perform poorly when a sufficient number of positive training examples are not available for each category (Prabowo and Thelwall 2009). To attain effective learning, 75 to 100 positive cases per category seem to be an absolute minimum (Beleites et al. 2013, Hopkins and King 2010). Due to the relatively high dimension of the injury report feature space (80 attributes) and the diversity of construction situations, the available training data were naturally sparse. For instance, in the report “carpenter felt discomfort in his left knee while exiting a tight area”, only 2 attributes are present, namely *existing/transitioning* and *confined workspace*. The other 78 attributes are not featured. Similarly, only a couple of attributes co-occur in each injury report. Therefore, for a large number of reports, cases when a given attribute is present are outnumbered by cases when this same attribute is not present. For example, in Desvignes’ (2014) data set of 1,280 manually-analyzed reports, the median attribute in terms of number of appearances, *heavy vehicle*, appeared only 21 times. Therefore, manually analyzing tens of thousands of injury reports would have been required to put together a satisfactory training database and achieve efficient learning.

Because such a large number of reports was not available, and because of time and resource limitations, we decided to design a NLP system based on hand-coded rules and dictionaries of keywords. Though this approach is not as simple and elegant, it offers some advantages over ML. Most importantly, it allows researchers to directly plug human intelligence and knowledge of the data into the system, allowing higher levels of accuracy to be reached (Sagae and Lavie 2003). As Wang et al. (2002) described, statistical classifiers are targeted at broad and relatively shallow understanding, whereas hand-crafted rules perform well within a specific domain when deep understanding is sought. Changes in coding scheme, detection of new variables, and higher skill can also be achieved very quickly by simply updating the rules and dictionaries, whereas algorithms require new, expensive training data to evolve and improve. Additionally, tools based on explicit rules avoid the somewhat opaque nature of ML models (Barbella et al. 2009, Breiman 2001a).

## Design of a rule-based natural language processing system

To develop our NLP tool, we capitalized on the operational definitions of variables, coding scheme, and knowledge of the data gained during the manual content analysis of more than 2,200 injury reports (Prades Villanova 2014, Desvignes 2014). Examples of injury reports from Desvignes (2014)'s database are provided in Table 3. Although these reports are not lengthy, they were written by personnel working on site within 8 hours of the injury as requested by the participating contractors' policies, and contain enough details to have a good idea of the work environment at the time of the accident and the injury outcomes. Other report examples can also be found in Table 5.

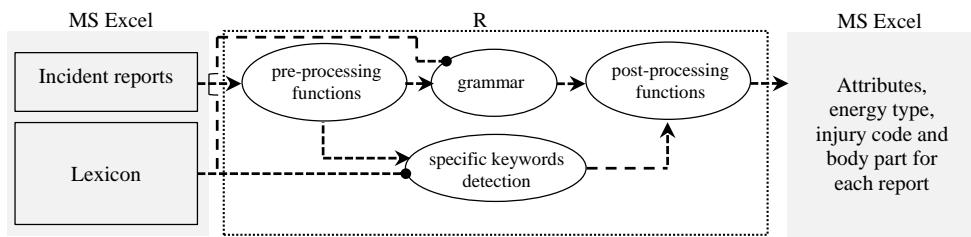
**Table 3. Examples of injury reports from Desvignes et al. (2014), with attributes and outcomes**

Reports	Attributes	Energy source	Injury code	Body part
A finisher apprentice was applying crystalline waterproofing to construction joints and cracks inside a pontoon cell. At some point the waterproofing material got in-between his kneepad and wet jeans which caused a concrete burn on his leg. The area was treated with a burn spray and he employee returned back to work immediately.	Hazardous substance, pontoon	Chemical	Exposure to harmful substance	Lower extremities
As employee was walking, he stepped on a nail that was lying in the road base/gravel. The nail was lying free on the ground. Employee felt the nail puncture his foot and immediately pulled it out and reported it to the safety representative who was nearby.	Nail, uneven walking surface, object on the floor	Motion	Struck by or against	Lower extremities
Employee was lifting a 2x12 wood plank when the wood plank got too heavy causing it to fall back towards the employee and hit him on the top/front of his hard hat.	Lumber, heavy material, lifting/pulling, improper security of materials	Gravity	Struck by or against	Head
At the ABC site wet tailrace a worker went to the Sea-Can for tools and PPE. When he went to open the door of the Sea-Can he received a 120 volt shock.	Door, electricity	Electricity	Exposure to harmful substance	Not detectable
A pipefitter was welding on a pipe support when slag fell into the cuff of his left glove, resulting in a burn to his left wrist.	Welding, steel sections, slag	Thermal	Exposure to harmful substance	Upper extremities

We built the system to automatically and accurately scan injury reports for the 80 validated attributes, 7 injury types, 5 body parts, and 9 energy types summarized in Tables 1 and 2. Some minor differences with Desvignes' (2014) original classification are to be noted: *object at height on same story* was grouped with the attribute *object at height*, and the precursors *slag* and *sparks* were separated. The attribute *snow/ice* was extended to include low temperature incidents (e.g., cases of hypothermia) and was renamed *adverse low temperatures*. The attributes *unpowered hand tool* and *powered hand tool* were extended to include tools that are not hand-held, but do not

belong either to the category *machinery* (e.g., girder spacer, power trowel, etc.). Therefore, these attributes were renamed *powered tool* and *unpowered tool*. Also, we added one attribute: *improper procedure/inattention*. Finally, we grouped the injury types *struck by* and *struck against* under the umbrella *struck by or against*, which is consistent with the Occupational Injury and Illness Classification System (OIICS) and BLS classifications (Hallowell 2008).

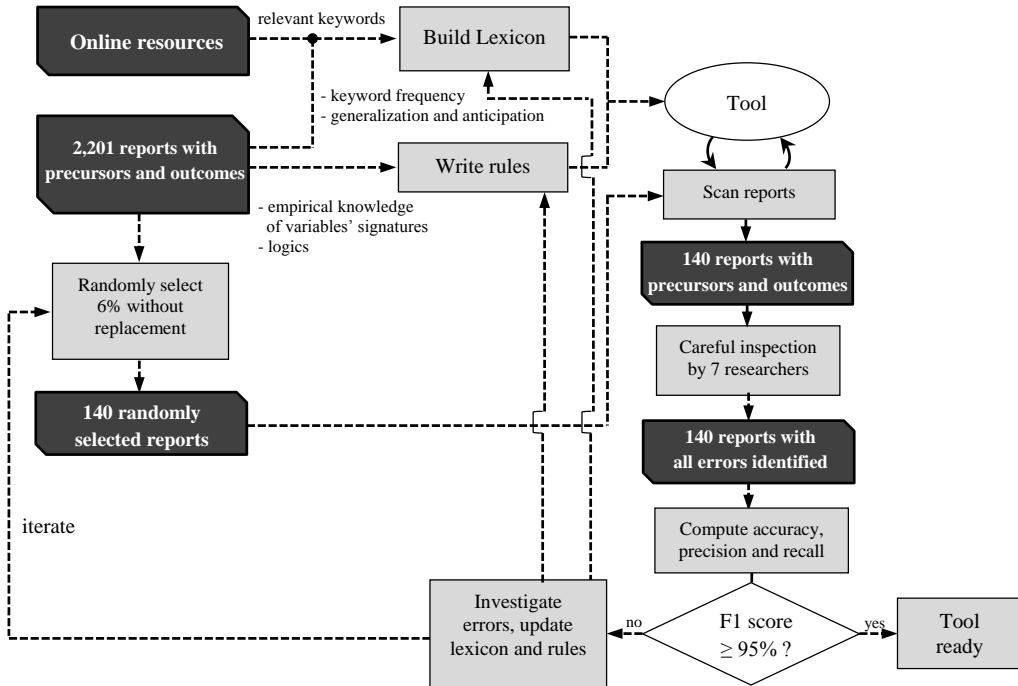
As shown in Figure 1, our NLP system consists of a catalogue of specific and generic terms (lexicon), and a set of structural rules allowing variables to be detected from the way terms are combined (grammar). Figure 2 describes the tool building and validation process. In what follows, details about the construction of the lexicon and grammar are given.



**Figure 1. Overarching tool process flow**

### **Step 1: lexicon building**

The first major step in the design of the automated content analysis system was the development of keyword dictionaries. Although there has been a great deal of construction safety research, no lexicon related to precursors of construction injuries was available at the time of this research. The lexicon we developed in this study is available upon request. To create it, several resources were leveraged as shown in Figure 2 and as discussed below.



**Figure 2. Tool building and validation process**

The first resource for this inquiry was the 2,201 manually analyzed incident reports from Prades Villanova (2014) and Desvignes (2014). These data include the attributes and outcomes shown in Tables 1 and 2 for each report, thereby allowing systematic sorting and identification of common keywords and phrases linked to each attribute. Online resources, such as the OSHA website, were subsequently used for dictionary enrichment. The focus of this section of the paper is on this decomposition process and the creation of the lexicon.

### **Specific keywords**

Using the R (R Core Team 2014) “tm” package (Feinerer et al. 2014), we obtained the most frequent terms associated with each variable. For example, some of the most frequent words associated with the energy type *chemical* were “burn”, “irritation”, “line”, “liquid”, “concrete”, “water”, “chemical”, “caustic”, “eye”, “acid”, “cloud”, “face”, “coker”, “laborer”, “burning”, “skin”, “insulation”, “insulator”, “mist”, “splatter”, “waterproofing”, “sprayed”, and “flushed.” Although only a simple frequency count, this list is very insightful. A

first observation can be made that some of the keywords in the list, such as “caustic”, “acid”, and “chemical”, are very specific to the energy type *chemical*. Indeed, the sole occurrence of these terms in a given incident report would suffice to classify the report as a *chemical* injury. Single keywords are known as unigrams (Manning and Schuetze 1999), and “caustic”, “acid”, and “chemical”, can thus be considered to be unigrams specific to the energy type *chemical*.

Within the context of *chemical* energy, the keyword “concrete” is an excellent example when additional words were required to derive meaning and to properly identify when concrete-related incidents were related to the chemical properties of the material. For instance, “concrete pouring” denotes a task; “concrete bag” a heavy material; “concrete drill” a powered tool; “concrete blanket” a type of tarpaulin; “concrete foreman” a person; and “concrete burn” refers to a type of chemical burn. Accordingly, although the sole presence of “concrete” or “burn” does not allow the variable *chemical* to be detected (i.e., “concrete” and “burn” are not unigrams specific to *chemical*), when these two words are found in a report as a pair (i.e., “concrete burn”), there is no ambiguity that the report deals with a *chemical* incident. Thus, “concrete burn” can be considered to be a specific double keyword (i.e., a specific bigram), associated with the energy type *chemical*.

We carefully inspected the most frequent terms for the 101 variables (i.e., 80 attributes, 9 energy types, 7 injury types, and 5 body parts), and gathered all unigrams, bigrams, and trigrams specific to each one. These terms were then stored in dedicated dictionaries (one per variable). In order to anticipate unseen cases, we used online resources for dictionary enrichment. For instance, synonyms of “caustic”, such as “corrosive”, “irritative”, “toxic”, etc., were looked up online and added to the dictionary for *chemical* even if these particular keywords did not appear in the reports available to our study. The OSHA website, for instance, contains a tremendous amount of valuable keywords covering a variety of topics. Some examples of the dictionaries of specific keywords included *steel sections* (37 unigrams, 80 bigrams, 2 trigrams), *lumber* (46 unigrams, 34 bigrams, 26 trigrams) and *poor housekeeping* (4 unigrams, 3 bigrams). Although specific keywords allow easy variable detection in some cases, it is often necessary to look at combinations of generic keywords, as explained next.

### **Generic keywords**

In many cases, variables have to be detected based on combinations of keywords that are not specific to any variable. We will call these terms *generic* keywords in what follows. Looking closely at the list of keywords for the energy type *chemical*, one can note that some keywords, such as “insulator”, “laborer”, “eye”, or “skin”, are related to persons, while some others refer to actions (“flushed”, “sprayed”), materials (“insulation”), outcomes (“irritation”), or location (a “coker” is an oil refinery unit).

None of these terms alone, if found in an injury report, would guarantee the presence of the variable *chemical*, nor of any other variable. However, *chemical* should be detected if “insulation” is associated with “eye”, “skin”, “irritation”, or “burning”. Therefore, adopting the anticipation and generalization process previously discussed, all the keywords related to the idea of irritation (e.g., “irritation”, “itching”, “burning”, etc.), all the keywords dealing with the human body (e.g., “skin”, “eye”, “hand”, etc.), and all the keywords about insulation (e.g., “insulation”, “fiberglass”, “glass wool”, “foam”, etc.) were collected and stored in dedicated dictionaries, and a grammatical rule for the detection of the energy type *chemical* was created.

Another example of an attribute requiring a detection rule based on generic keywords is *working at height*. This precursor should be detected when a term linked to the topic of working (e.g., “working”, “doing”, “performing”, “drilling”, “installing”, etc.) is associated with a term involving the notion of height (e.g., “height”, “elevation”, “elevated”, “mezzanine”, “roof”, etc.). These simple examples are provided for illustration purposes, but in practice, relying simply on association was not enough to avoid false alarms. Detection rules had to take into account context, that is, term order, text structure (punctuation, prepositions, conjunctions, etc.) and the presence or absence of keywords related to other concepts. The rule building process will be described in detail in the following section.

Since generic keywords do not allow specific variables to be detected until they are combined with other keywords, generic keywords were not organized in variable-related dictionaries, but rather, stored in topic or

concept-related dictionaries. Some of the final 47 dictionaries were, for example, *people* (141 unigrams, 8 bigrams, plus the 475 most given first names in the U.S.), *working* (117 unigrams, 1 bigram), *elevation* (21 unigrams, 1 bigram, 1 trigrams), and *unstable* (47 unigrams, 24 bigrams).

Note that the notions of specific and generic terms originated in the field of linguistics. For example, i Cancho and Solé (2001) observed, using graph theory, that there are two different regimes of words: basic and specialized. Also, it is important to keep in mind that dictionary creation is an iterative process. No lexicon is fully comprehensive and careful inspection of the system's errors, deep understanding of the textual data, and full use of available linguistics resources are required to ensure a continuous improvement of the dictionaries (Grimmer and Stewart 2013). Such updating, while time-consuming and sometimes tedious, is necessary (Kuechler 2007). As shown in Figure 2, and as will be discussed, errors were closely examined during the validation process, and keywords and rules were tuned accordingly.

### **Step 2: Devising detection rules**

The second step in the development of the system was the writing of grammatical rules to detect variables based on combinations of generic keywords. This task was complex. Simply testing for co-occurrence of topics was not sufficient and created many errors. To illustrate, consider the following two injury report excerpts:

Excerpt 1: “**moving floor plate**, employee strained his back.”

Excerpt 2: “employee rolled his ankle on **moving floor plate**.”

In each excerpt, the generic keywords in bold are associated with the same topics and appear in the exact same order. Yet, very different variables should be identified in each case. In the first report, it should be inferred that a back injury was sustained as a result of the presence of the attribute *material handling*, whereas in the second report, the worker sprained their ankle due to the occurrence of the precursor *unstable support/surface*. The

ambiguity comes from the fact that the bigram “floor plate” has two different senses: (1) material and (2) walking surface. If “floor plate” refers to some material, then it is being moved (necessarily by a person) and “moving” is a verb. On the other hand, if “floor plate” designates a walking surface, then it is moving as a result of an unintended action or some exterior influence and “moving” is used as an adjective to qualify “floor plate”.

To detect the correct attribute, it is necessary to determine which of the senses of “floor plate” is invoked in each report, or in other words, to disambiguate the sense of the generic bigram “floor plate”. In many instances, disambiguation is enabled by looking at the context, such as looking at preceding and following words. Also, taking into account prepositions such as “on”, “onto”, “into”, “under”, and “over” gives a lot of information. Finally, the structure of the report can be decomposed by using punctuation marks such as commas and periods, and conjunctions such as “and”, “or”, and “while”. In the given example, it should be noted that the attribute *unstable support/surface* can be rightfully identified in the second report if the fact that the preposition “on” precedes “floor plate” is captured.

26 variables were detected based on their specific keywords only. For the energy type *biological*, for instance, the occurrence of “biosludge”, “scorpion”, “bees”, “bugs”, or “wastewater” (and others very specific, unambiguous keywords) was enough to trigger detection. However, for 76 of the 101 variables, it was necessary to account for the subtleties previously mentioned, namely (1) term ordering, (2) report structure (punctuation, prepositions, conjunctions), and (3) presence or absence of terms related to other concepts. As shown in Figure 2, the signature of each variable was empirically determined from knowledge gained during the manual content analysis of more than 2,200 reports, from generalization, anticipation, and grammatical logic; and from lessons learned during the tool tuning process.

Of the 76 variables requiring grammatical rules, 51 were detected based on a combination of specific keywords and rules, and 25 did not have specific keywords and were detected based on rules alone. An example of an attribute with mixed rules was *confined workspace*, where “potholing”, “manhole”, “tunnel”, and others were

used as specific keywords. To capture all other cases, the variable was detected if any element from the topic *confined* (e.g., “confined”, “limited”, “tight”, “narrow”, etc.) was found combined with any element from the family *area* (e.g., “area”, “room”, “space”, “quarters”, “entrance”, etc.).

The remaining 25 variables were detected based on rules alone. For instance, the grammatical rule for the attribute *object at height* can be written in English as “(ANY element from the topic *materials.tools* is followed by any element from the topic *fall*) AND (NO element from the topic *people* is found sandwiched between ANY element from the topic *materials.tools* and ANY element from the topic *fall*) AND (ANY element from the topic *elevation* is present). For *cable*, the rule is much simpler, and can be written in English as “ANY of the keywords (“cable”, “cables”) is present, but is NOT immediately followed by ANY of the keywords (“tray”, “shovel”, “wheel”, “reel”, “spool”, “coil”, or “trench”). Indeed, *cable tray* is another attribute; a cable shovel is classified as a *heavy vehicle*; wheels, reels, coils and spools are categorized as *spool*; and “cable trench” is not specific to *cable* (e.g., “carpenter tripped on cable trench”). Note that this is done without loss of generality, since both “cable” and “cable trench” are still free to co-occur. In the sentence “worker was installing cable in cable trench”, *cable* would still be detected.

To efficiently write these statements in the R programming language, we developed a library of custom functions, which we introduce next.

### ***Creation of R functions***

Table 4 shows the R functions we created for rule-writing. Like for the lexicon, these functions can be made available on a per-demand basis. Every single rule was written as a combination of statements involving these functions. As shown in Figure 1, a necessary first step was to preprocess unstructured text. To this purpose we used functions from the “base” (R core team 2014) and “tm” (Feinerer et al. 2014) R packages.

More precisely, preprocessing included (1) lower case conversion (R is case sensitive), (2) partial punctuation removal, (3) custom stopword removal, (4) stripping extra white space, and finally (5) splitting text based on whitespace (or “tokenizing”). In removing punctuation (step 2), commas and periods were kept, since they provided valuable information about text structure.

For example, in: "accident involved one welder. Falling hammer from mezz deck struck him", the immediate proximity between the two keywords "welder" and "falling" should be disregarded because these two words do not belong to the same sentence. Hence, the machine should understand that the welder himself is not falling.

The third step, stopwords removal, is standard and used to some degree in all text mining applications (e.g., Pereira et al. 2013, Caldas and Soibelman 2003). We removed stopwords such as "what", "too", "a", "be" and others, but kept words referring to persons, such as "she", "he", "they", "his", "her", etc. as well as other words like some prepositions and conjunctions (e.g., "below", "between", "into", "and", "so", etc.), because they proved very useful. We also kept numbers and intra-word dashes, since a number of keywords included such characters. For instance, some specific unigrams for the attribute *lumber* were "2x2", "2x6", and some for the precursor *bolt* were "she-bolt" and "u-bolt".

Finally, after stripping the extra whitespace (step 4), a fifth and final step consisted in splitting text based on whitespace. This action, known in the text mining field as tokenization, turned unstructured text into an ordered character vector. The elements of the vector were words, kept punctuation marks, and numbers. Each element was assigned an index number, indicating its position in the vector. This step was fundamental, as it allowed distance between two elements, (in terms of the number of other elements separating them), to be measured. This intuitive distance will be referred to as "radius" in what follows.

The key assumption when working with radii is that only words close to each other are related. Beyond a certain distance, the dependence between words fades until the words are not related anymore. This is consistent with the

Markov assumption in NLP, which holds the local context of a word only to be of importance (Manning and Schuetze 1999). Put differently, “You shall know a word by the company it keeps” (John R. Firth 1957:11). An equivalent paradigm is followed in time series analysis: the correlation between observations decreases and eventually disappears as temporal distance increases (Wei 1994, pp. 6-26).

**Table 4. Comprehensive list of functions used for writing rules**

FUNCTION	RETURNS TRUE IF... (FALSE ELSE)	ORIGIN
statement1   statement2	(statement 1 is true) or (statement 2 is true)	
statement1 & statement2	(statement 1 is true) and (statement 2 is true)	
any(a %in% x)	any unigram of the character vector* a is present in the character vector x	R base package
any(sapply(a, grepl, text))	any unigram, bigram, or trigram of the character vector a** is present in the text	
complex.s(a, b, radius, text)	any element of the character vector a is followed by any element of the character vector b in the text, in the same part of the sentence, and within the radius provided.	
tricky.double(a, b, c, number, radius, text)	<ul style="list-style-type: none"> <li>• If number=1 any element of the character vector a is present in the text, but is not followed by any element of the character vector b within the radius provided,</li> <li>• If number=2 any element of the character vector b is present in the text, but is not preceded by any element of the character vector a within the radius provided,</li> <li>• If number=3 any element of the character vector b is present in the text, but is not preceded by any element of the character vector a within the radius provided, nor followed by any element of the character vector c within the radius provided</li> </ul>	developed in R in this research
sandwich.wrap(a, b, c, text)	any element of the character vector c is not sandwiched between any element of the character vector a and any element of the character vector b	

\*the term “character vector” simply refers to an ordered vector of elements. The character vectors *a*, *b*, and *c*, are used to represent the content of specific and generic keywords dictionaries. The order in which the elements appear in the dictionaries does not matter. On the other hand, the character vector *x* is the text of the injury report split based on whitespace (last preprocessing step previously described), so the order matters and corresponds to the order in which elements appear in the text.

\*\* in this case, the elements of *a* have to be regular expressions

The functions summarized in Table 4 enable the writing of rules capturing complex word dependency and word order signatures. These subtleties are typically not taken into account by traditional ML approaches, based on the “bag-of-words” construct, such as the standard TFxIDF-SVM approach of Joachims (1998).

### **Illustrative Example**

An example of rule building using the R functions previously introduced will be presented. These examples consider the incident report: “the employee was walking down the stairs and slipped”. After being cleaned and structured into an ordered vector of elements, this report can be represented as shown in Figure 3. If the goal is, say, to detect the attribute *slippery walking surface*, one rule can be written in English as “ANY element corresponding to the idea of *people* is followed by ANY element from the topic *slipping* WITHIN a radius of 7”. This rule can be translated in R as a single **complex.s()** statement, and is capable of detecting the variable *slippery walking surface*, as illustrated in Figure 4. However, caution should be used when working with radii. If the distance prescribed is too short, variables can go undetected (a radius of 3 in the example at hand, for example). However, if the distance prescribed is too long, the risk of capturing spurious relationships and thus of improperly detecting variables is increased, as shown in Figure 5. In order to minimize the risk of such false alarms, the rule can be modified as follows: “(ANY element corresponding to the idea of *people* is followed by ANY element from the topic *slipping* WITHIN a radius of 7) AND NOT (ANY element of the topic *slipping* is preceded by ANY element associated with the concept of *tools* WITHIN a radius of 3)”. This rule can be written in R using two **complex.s()** statements, as shown in Figure 6. With such a new rule, the false alarm is avoided.

Furthermore, because only *persons* and (*materials* or *tools*) can slip, the rule can be simplified as: “ANY element from the topic *slipping* is present, but this element is NOT preceded by ANY element from the topic *materials.tools* WITHIN a radius of 3”. This can be written in R as a single **tricky.double()** statement. Some other noteworthy examples of when **tricky.double()** statements prove useful include “line of fire” (*thermal* should be detected when “fire” is present but not when “fire” is immediately preceded by “line of”), “chain fall”, (a chain fall is a tool and has nothing to do with falling), “concrete vibrator” (*concrete* should be detected when “concrete” is present but not when it is immediately followed by “vibrator”, “hammer”, etc.), “rebar foreman” (*rebar* should be detected when “rebar” is present but not when it is immediately followed by “foreman”, “finisher”, etc.), “chipping hammer” (*chipping* should be detected when “chipping” is present, but not when it is

immediately followed by “hammer”), or “finger nail” (*nail* should be detected when “nail” is present, but not when it is immediately preceded by “finger”).

### ***Processing injury reports***

When presented with reports, as shown in Figure 1, the NLP tool starts by selecting the first report. This report is cleaned, and scanned for any specific keyword associated with any variable. Then, detection rules for all variables are tested. When all attempts to detect variables have been made, a binary vector of length the total number of variables is returned. The binary vector features “1” whenever the corresponding variables have been detected, and “0” elsewhere. These steps are repeated for all reports. Because each report is scanned independently, parallel processing could be used to speed up the process. For that purpose, the “foreach” and “doParallel” R packages (Revolution Analytics and Steve Weston 2014) were used. By using 36 cores at 2.6GHz (compute-optimized Amazon EC2 instance), 4,377 reports could be analyzed in just under 11 minutes. Optimization could lead to much faster performance. After the tool is done looping through all the reports, a binary matrix is obtained, as illustrated in Figure 7. The following step consists in resolving conflicts.

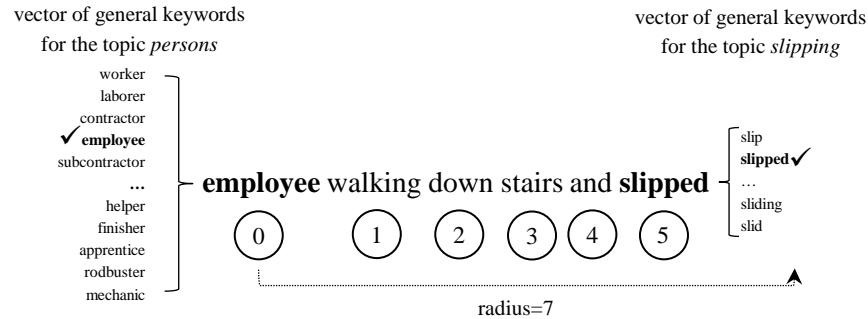
### ***Resolving conflicts among detected variables***

We created clash detection and resolution rules centralized within a conflict resolution function. This function served as an internal control in the coding structure. Indeed, some variables are incompatible, such as *fall on same* and *fall to lower level*, while some others are implicitly connected, like *radiation* and *exposure to harmful substance*. After a given report has been scanned, it is thus necessary to make sure that the variables that have been detected are not incompatible and that no natural association has been missed.

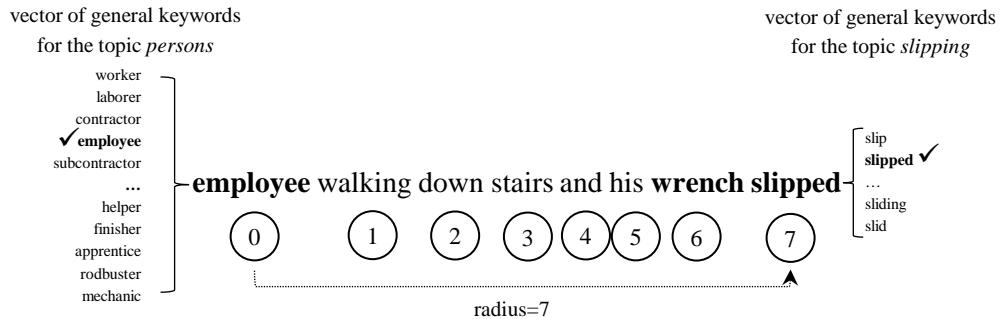
### employee walking down stairs and **slipped**



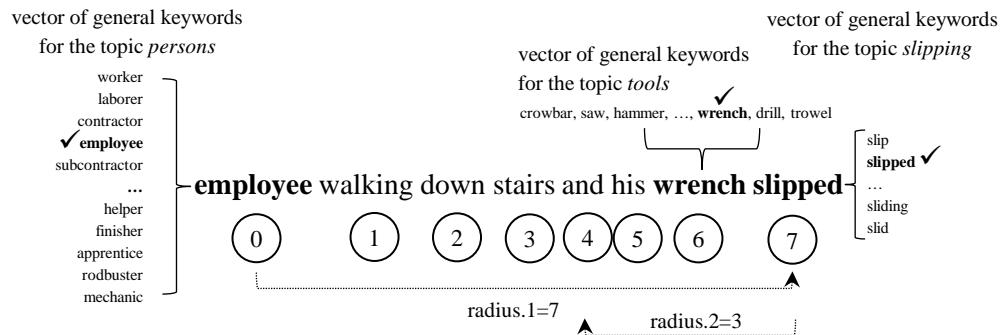
**Figure 3. Cleaned and structured incident report**



**Figure 4. Detection of the variable *slippery walking surface* based on a single complex.s() statement**



**Figure 5. Faulty detection of the variable *slippery walking surface* by a single complex.s() statement**

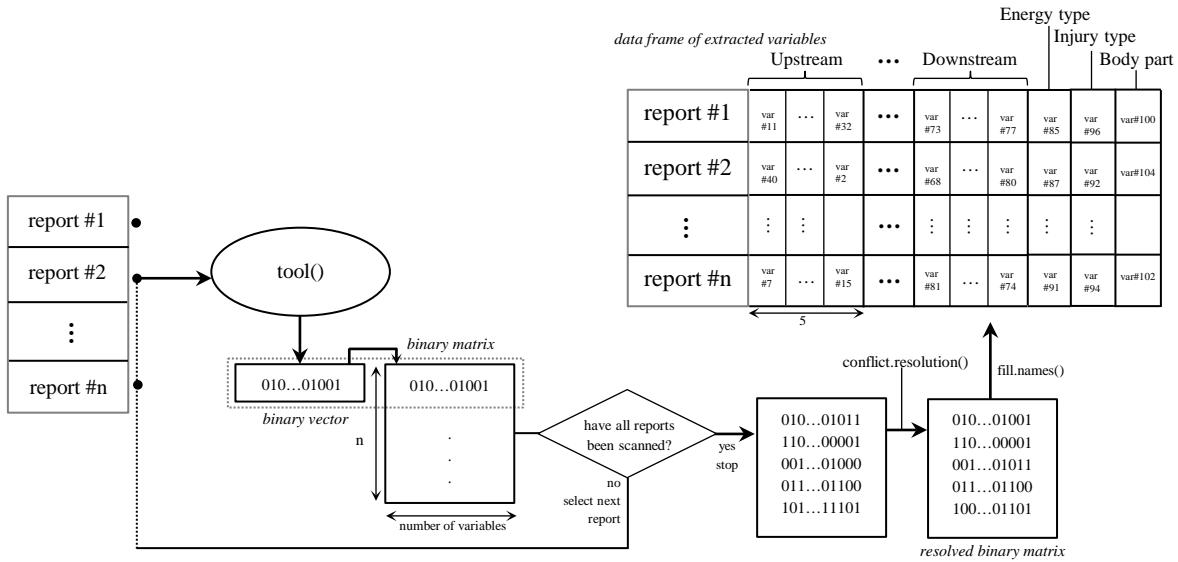


**Figure 6. Two complex.s() statements allow the false alarm to be avoided**

For example, if the keyword “ice” is present, the attribute *ice* will be detected (except in cases where ice packs are applied on bruises, ice buckets are carried, etc.). But because topics about *persons* and *slipping* will always be present in ice-related incident reports, and moreover, combined in the same fashion (“employee slipped on [...]”), the attribute *slippery walking surface* will always be detected if *ice* is detected. While not fundamentally incorrect (*ice* is indeed a subset of *slippery walking surface*), this association is problematic. In fact, the attribute-based framework strives to produce high quality structured data on which predictive statistical modeling can be successfully applied. For that purpose, overlaps in the attributes are to be avoided as much as possible to ensure that every precursor keeps its full predictive power. With respect to the aforementioned example, the conflict resolution rule consisted in deleting *slippery walking surface* whenever *ice* was also detected. Examples of similar conflict resolution rules included *sparks* and *small particles* (*sparks* was preferred), *stairs* and *ladder* (*ladder* was preferred), *fall on same level* and *fall to lower level* (*fall to lower level* was preferred).

In addition to rules precluding redundancies, other rules were developed to ensure proper association. Indeed, some variables should always be found together. For example the injury type *struck by or against* should always be associated with the energy type *motion*. In the same manner, the presence of the energy types *electricity*, *thermal*, or *chemical*, should always trigger the detection of the injury type *exposure to harmful substance*. Finally, if the injury types *fall to lower level* or *fall to same level* have been detected, the energy type *gravity* should always be added.

As described in Figure 7, when all conflicts have been resolved by the `conflict.resolution()` function, the binary matrix is converted to a textual matrix by the `fill.names()` function. This matrix comprises injury reports as rows, and the attributes and outcomes detected as columns (see Table 5). Both matrices (binary and textual) are automatically written by the tool as MS Excel spreadsheets, and can also be analyzed directly in R.



**Figure 7. Scanning of injury reports and post-processing**

**Table 5. Example of the system's output for 12 injury reports**

Description	up.1	up.2	up.3	trans.1	trans.2	trans.3	down.1	energy	code	body part
Employee was welding on a pipe, as he brought hands down he touched the tungsten with left finger resulting in a burn.	piping	welding						thermal	exposure to harmful substance	upper extremities
The employee was in the process of hoisting a piece of cable tray to an above level and he scraped his arm on a sharp edge of the cable tray.	cable tray			unpowered tool	lifting pulling manual handling	sharp edge		motion	struck by or against	upper extremities
Climbing out of scaffold and felt back discomfort.	scaffold	working at height		exiting				motion	overexertion	trunk
Employee was grinding a pipe in a tight spot, grinder kicked back making contact with right thumb resulting in an abrasion.	confined workspace	grinding	piping	powered tool				mechanical	struck by or against	upper extremities
EE was lifting a 2 X 12 wood plank when the wood plank got too heavy causing it to fall back towards the EE and hit him on the top/front of his hard hat.	heavy material/tool	lumber		lifting pulling manual handling			improper security of materials	gravity	struck by or against	head
Welding FOB	welding			small particle				motion	struck by or against	head
Employee was offloading burners with a cart when the cart moved unexpectedly "crushing" his leg between the cart and the existing railing.	unpowered transporter	guardrail/handrail		lifting pulling manual handling				motion	caught in/compressed	lower extremities
Employee was grinding overhead, weight shifted and board he was standing on slid, causing a "pop" to upper leg resulting in a strain.	grinding	lumber			working overhead	unstable support/surface		motion	overexertion	lower extremities
勞動者在一次破磚頭的操作中受傷。	chipping							chemical	exposure to harmful substance	not detectable
The employee reported that he received an insect bite/sting on 6/21/13 at work.				insect				biological	exposure to harmful substance	not detectable
Employee was walking out of the warehouse building. The floor was wet from the rain. He slipped and fell on his left knee on the concrete floor.	concrete	slippery walking surface		exiting				gravity	struck by or against	lower extremities
An employee of --- Mechanical was using Oxy/Acetylene to cut a pipe when hot slag entered his sleeve and burned his wrist which was treated on site. Cutting gloves were not being used.	piping	welding		slag			no or improper PPE	thermal	exposure to harmful substance	upper extremities

## **Validation of the system and measurement of reliability**

As shown in Figure 2, we adopted an iterative process to tune the tool. At each step, 140 injury reports were randomly drawn without replacement from the data set of 2,201 reports of Desvignes (2014) and Prades Villanova (2014). These reports were then automatically scanned by the R system, and as previously explained, a textual matrix of 140 rows by 19 columns (1 column for the reports, 5 for upstream attributes, 5 for transitional attributes, 5 for downstream attributes, 1 for energy type, 1 for injury code, and 1 for body part) similar to Table 5, was returned. This output was divided into 7 textual matrices of 20 rows each. Each table was assigned to a researcher, who had been involved with the Desvignes (2014) study and was familiar with the coding scheme and operational definitions of variables. As illustrated by the retrieval matrix in Table 6, each researcher reviewed their randomly assigned piece of output looking for true positives (TP), false positives (FP) and false negatives (FN).

**Table 6. Retrieval matrix (adapted from Buckland and Grey 1994).**

		Not relevant	
		Relevant	Not relevant
Retrieved	Relevant	TP	FP
	Not retrieved	FN	TN

True positives (TP), also informally called “hits”, refer to cases when the tool has rightfully detected a precursor that was indeed present in the injury report. False positives (FP), also called “false alarms”, or type I error, designate an instance when the tool has wrongfully detected an attribute that was not present in the injury report (not relevant). Finally, false negatives (FN), also called “misses”, or type II error, are cases when the tool has omitted to detect the presence of a precursor that was actually there in the report (not detected and relevant). It should be noted that the last option, true negatives (TN), occurs when the

tool has not detected an attribute that was not present in the injury report. True negatives were not taken into account.

After careful examination by the seven researchers, the reviewed textual matrices were aggregated and all true positives, false positives, and false negatives were counted. Three performance metrics, standard in the field of information retrieval and NLP, were then computed: precision, recall and F-1 score (Al Qady and Kandil 2014, Yu and Hsu 2013, Buckland and Grey 1994). It should be noted that inspection of the system's output agreed upon by seven humans was preferred over automatic comparison to a gold standard (e.g., manually labeled reports of Prades Villanova 2014 and Desvignes 2014). Indeed, this allowed the source of each error to be tracked back and fixed by tuning the grammatical rules and lexicon accordingly. This output examination process played a crucial role in reaching the targeted accuracy.

Precision was calculated, as shown in equation 1, as the proportion of relevant items to the number of items detected (Buckland and Grey 1994). This is simply the probability that an attribute is present given that it was detected by the tool (Goutte and Gaussier 2005). High precision means that most results returned are relevant. Maximum precision is attained in the absence of type I error (i.e., no false positive). On the other hand, recall is the number of retrieved relevant items as a proportion of all relevant items (see equation 2). In other words, recall is the probability that a precursor that is present is detected by the tool. A high recall rate means that most of the relevant results are returned, and recall is maximized in the absence of misses (i.e., in the absence of type II error). Buckland and Grey (1994) define precision as the purity of retrieval and recall as the completeness of retrieval.

$$\text{precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$

**Equation 2. Recall**

$$\text{recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

### **Equation 1. Precision**

More precisely, the formulas that we used macro-averaged the results (Prabowo and Thelwall 2009). Instead of considering recall and precision rates for each variable separately (what is known as micro-averaging), true positives, false positives, and false negatives for all categories were aggregated and the recall and precision rates averaged these counts. Macro-averaging treats each class equally, and is harsher than micro-averaging since one class that results in a bad performance can significantly deteriorate the overall performance (Prabowo and Thelwall 2009).

Finally, the F-1 score is defined as the harmonic mean of precision and recall. As shown in equation 3, the F-1 score gives the same weight to precision and recall, assuming that the cost of a false positive equals the benefit of a true positive.

$$\text{F1 score} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

### **Equation 3. F-1 score**

A threshold of 95% in F-1 score was selected *a priori* as a tool tuning stop criterion. This is a high threshold, especially when using strict macro-averaging performance metrics, but it was crucial to show that our NLP program was able to reach the same performance as human coding. The scores for each round of random reviews are summarized in Table 7. Four iterations were needed before the threshold was achieved. As shown in Figure 2, lessons learned from careful examination of the errors made by the system played a huge role in improving skill.

The error rates for energy type and injury type, simply defined as the number of errors divided by the number of reports scanned ( $n=140$ ), were also computed at each round. For body part, to avoid any false alarm, the tool was designed to return *not detectable* when more than one body part is detected, or when the information is not present in the report. The detection of body part was only added to the system's functionality after the third iteration of random reviews, and *not detectable* was returned 6.25% of the time. When any body part was detected, the system was correct every time.

**Table 7. System's performance at each step of the tuning process**

iteration	Precision (%)	Recall (%)	F1-score (%)	Error rate for energy type (%)	Error rate for injury code (%)	“Not detectable” rate for body part (%)
1	85.4	92.4	88.8	11.5	8.0	NA
2	91.1	94.6	92.8	8.6	4.3	NA
3	91.7	95.9	93.8	3.6	2.9	NA
4	95.0	97.0	96.0	5.7	5.7	6.25%

These scores are comparable or even better than the scores attained by most statistical classifiers found in the literature. For instance, Verma et al. (2011) used a naïve Bayes classifier to analyze tweets and reached 80% accuracy. Tweets were to be classified into 5 categories. A final accuracy of 0.65 with a recall rate of 0.75 was obtained by Grimmer and Stewart (2013), with a Random Forest classifier. Go et al. (2009) used unigrams and bigrams as variables to classify tweets as positive or negatives and an accuracy of 83% was reached. Finally, Bai (2011) reviewed two studies about automated opinion mining and movie review polarity categorization and reported accuracies ranging between 66% and 88.9%.

In the construction field specifically, Al Qady and Kandil (2014) implemented unsupervised clustering algorithms to classify project documents into mutually exclusive classes and reached a 0.844 F-score in the optimum case. The computer aided drawings automated retrieval system of Yu and Hsu (2013) attained 100% recall, but the precision was only 57.2%. Finally, 86.37% accuracy was reached by Caldas

and Soibelman (2003), who used Support Vector Machines to classify construction project documents into hierarchical classes.

It is not surprising that our tool based on handcrafted rules compares favorably to the aforementioned machine learning (ML) models, since as noted by Sagae and Lavie (2003), hand coded rules allow researchers to transfer their expertise, knowledge of the data, and human intelligence to the system. Tools based on manually devised rules are usually capable of deeper understanding than most ML models when the domain of application is very specific (Wang et al. 2002). For instance, the rule-writing functions developed in this research allow complex patterns of word dependency and word order signatures to be captured, which is not possible with traditional ML techniques that are based on the “bag-of-words” approach.

## **CONCLUSION, LIMITATIONS, AND RECOMMENDATIONS**

Major construction firms and federal agencies have been recording injury-related events and near misses in the form of digital textual reports for many years, but due to the lack of an adapted framework and methodology, and because manual content analysis is fraught with time, labor, and organizational limitations (not to mention inter-coder reliability issues), the greater part of this valuable knowledge has been left unstructured and unexploited so far. Specifically, low severity, high frequency events that are not OSHA-recordable but account for huge financial and long term human costs are typically not investigated (Hinze et al. 2006).

In this study, we tested the proposition that the needs for manual content analysis of incident reports can be eliminated using NLP. Results clearly show that this is possible. Indeed, the R system we developed proved capable of scanning naturally occurring, unstructured textual injury reports for 101 relevant, valid and carefully defined variables (80 precursors, 7 injury types, 9 energy types, and 5 body parts) with high

recall (0.97) and precision (0.95) rates. As will be further discussed in the recommendations section, the proposed NLP system will enable organizations to quickly and automatically extract the knowledge contained in their unstructured injury report databases to improve safety management.

## **Limitations**

First, our system is inherently limited by the use of hard detection rules: it is not robust to unfamiliar and erroneous input, such as misspelled, missing, and unseen words. In other words, the system cannot address situations that were not anticipated, and the quality of the available textual data directly affects the quality of the attribute and outcome data extracted by the system. For instance, when faced with a misspelled word (e.g., “steal” instead of “steel”) our tool is unable to detect the associated variable *steel/steel sections*. Most frequent misspellings can be anticipated, but obviously, it is unfeasible to account for all potential cases. Also, some reports contain a description of the events following the incident (e.g., “[...] worker was brought to the job trailer and an ointment was applied”). Despite the many precautions we took, certain precursors can still be wrongfully detected from these irrelevant portions of the text.

Our system reached very high scores during random reviews of its output, indicating that the impact of erroneous or misleading input was very limited. Indeed, the reports available to this research were generally carefully written, and only contained facts relevant to the incident. However, verifying the quality of the injury reports is a necessary first step that should always be taken before running our tool on any new database in the future. Reports of overly poor quality may prevent the automated content analysis from being successfully conducted.

The methodology introduced in this study is applicable to other domains. Especially, the system structure (i.e., the overall approach, methodology and rule-writing functions that were developed) is ready to be

used in any situation. Also, the full system as it is ready to scan incident reports pertaining to the industrial, energy, infrastructure, and mining fields, since the grammatical and conflict resolution rules as well as the lexicon were tuned and validated on such reports. If scanning reports belonging to other industry sectors is desired, a prerequisite for reaching high skill will be to extend the tool's lexicon and detection rules. Defining new fundamental attributes may also be needed, which requires trained content analysts and calibration meetings. As Grimmer and Stewart (2013) note, a limitation of dictionary-based methods is that they are only efficient inside the domain for which the dictionaries were originally developed. Finally, tuning the tool (i.e., creating/updating rules) requires basic R literacy.

### **Recommendations for future research**

When a sufficient amount of reports have been classified by the R system in each category, a natural next step would consist in hybridizing the system with ML algorithms such as Support Vector Machines. As a hybrid, the system would keep the deep understanding given by hand coded rules, while acquiring the flexibility of statistical classifiers. Moreover, by judiciously aggregating the decisions of hard and soft detection rules, the skill of the system could increase even more (Prabowo and Thelwall 2009).

In order to get closer to 100% accuracy, the errors made by the tool can be automatically detected using data mining methods such as hierarchical clustering, which is known for its ability to isolate outliers in small clusters. Manually inspecting these small clusters allows easy identification and correction of the errors. This approach can be used for instance as a post-processing step in order to attain maximal signal over noise ratio before training statistical predictive models on the data.

The main contribution of the proposed NLP system is its ability to extract meaningful structured attribute and outcome data from unstructured injury reports automatically and with high accuracy. Such structured data represents raw material required to apply data mining and statistical modeling algorithms that will

allow to better understand, predict, and prevent the occurrence of construction accidents. For instance, models predicting safety outcomes from combinations of attributes will assist safety managers in accurately diagnosing the safety risk associated with specific construction situations and provide them with tailored recommendations based on simple observations of the work environment. The proposed system can be seen as a construction safety knowledge discovery tool: large databases of injury reports represent a wealth of valuable lessons waiting to be learned and made readily available to construction professionals in order to help decision-making and to prevent mistakes from being repeated over and over.

The methodology presented in this research can be applied to mine other construction textual data like contracts and project documentation, but more generally, it can be applied to mine any kind of text. The use of NLP may soon become mandatory and widespread in construction management to make sense of the ever-growing amount of digital information associated today with even the smallest construction project. Statistical modeling and ML algorithms are without a doubt the present and the future of text analytics, but this study shows that when very high accuracy is sought on a specific, well-defined domain, dictionaries and hand-coded rules can bring satisfying results, once a one-time upfront investment has been made.

**CHAPTER 3: SAFETY CLASH DETECTION FOR BIM, WORK  
PACKAGING, AND OPERATIONS: IDENTIFYING SAFETY  
INCOMPATIBILITIES AMONG FUNDAMENTAL CONSTRUCTION  
ATTRIBUTES BY APPLYING GRAPH MINING AND HIERARCHICAL  
CLUSTERING TO ATTRIBUTE DATA SETS**

## **ABSTRACT**

Construction still accounts for a disproportionate number of injuries, inducing consequent socioeconomic impacts. Despite recent attempts to improve construction safety by harnessing emerging technologies and intelligent systems, most frameworks still consider tasks and activities in isolation, and use secondary, aggregated, or subjective data that prevent their widespread adoption. To address these limitations, a newly introduced conceptual framework and accompanying natural language processing system was used to extract standard information in the form of fundamental attributes from a set of 5,298 raw accident reports. State-of-the-art data mining techniques were then applied to discover attribute combinations that contribute to injuries. These incompatibilities were referred to as construction safety clashes. The main contribution of this study lies in the methodological advancements that it brings to the construction safety domain. In light of the results obtained, the approach shows great promise to become a standard way of extracting valuable, actionable insights from injury reports in a fully unsupervised way. The use of this methodology could enable construction practitioners to ground their safety-related decisions on objective, empirical data, rather than on limited personal experience or expert opinion, which is the current industry standard. Finally, the methodology allows construction accidents to be viewed as perturbations in underlying networks of fundamental attributes. While the analysis of the current data set provides preliminary evidence for this theory, comparison to non-accident reports will be required for validation.

## **INTRODUCTION AND MOTIVATION**

Even though safety performance has notably improved after the inception of the Occupational Safety and Health Act (OSHA) of 1970, construction fatalities, disablements, and illnesses still have a dramatic socioeconomic impact. In fact, construction still accounts for a fatal occupational injury rate of 9.4 per 100,000 full-time workers, one of the highest in the United States. (Bureau of Labor Statistics 2013). Moreover, the construction industry has consistently accounted for the most fatalities of any industry in

the private sector since 2005, with 796 casualties in 2013 alone. Therefore, improving safety has become an absolute priority.

Construction has reached saturation with respect to the traditional safety strategies that were originally implemented to comply with regulations (Esmaeili and Hallowell 2011a). Therefore, safety researchers and professionals have recently tried to harness emerging technologies and intelligent systems that are traditionally used for design, planning, or operations. Some examples of such technologies include Building Information Modeling (BIM), proximity sensing, or information retrieval. While these efforts are worthy, they currently suffer limitations, as the data used are mostly secondary, aggregated, and subjective (based on regulations, intuition, or judgment); and tasks are considered in isolation, preventing the efficient capture of the transient and dynamic nature of construction work (Prades Villanova 2014).

To improve the robustness of safety analyses, Esmaeili and Hallowell (2012, 2011b) and Esmaeili (2012) introduced a conceptual framework where any injury can be characterized by a unique combination of universal context-free descriptors of the work environment, also called fundamental attributes or injury precursors. These works made great strides by showing possible the extraction of objective, standardized structured information from unstructured injury reports, opening the gate for the first time to leveraging big, empirical, and objective safety-related data. However, several major limitations remained, such as the needs for a more comprehensive set of attributes and for an automated system to scan the reports. Prades Villanova (2014) and Desvignes (2014) addressed the first limitation by proposing a refined and expanded list of fundamental attributes, and Tixier et al. (2016a) addressed the second one by developing a highly accurate (96% in F1 score) natural language processing (NLP) system.

In this study, we tested the extent to which graph mining and hierarchical clustering can be used to identify safety-critical associations of attributes from large data sets. We conducted our experiments on an

attribute data set obtained from scanning 5,298 raw injury reports with Tixier et al. (2016a)'s NLP system.

## **BACKGROUND AND POINT OF DEPARTURE**

This study was built upon a foundation of knowledge in two key areas: construction safety analysis, and safety integration in BIM. Although both of these areas have received some attention from the scientific and practical communities, researchers have yet to explore their nexus. The following literature review highlights current limitations in both domains and develops a firm point of departure.

### **Construction safety risk analysis**

Safety analysis in construction has taken many forms and varies greatly in the *data sources* used, and the *level of detail of the units of analysis*.

#### ***Data sources***

The vast majority of construction safety studies rely on opinion-based risk data, generally obtained by asking experts to rate the relative magnitude of risk based on their professional experience and intuition (Prades Villanova 2014). Such data are subjective and suffer the numerous biases that affect human judgment under uncertainty, such as overconfidence, anchoring, availability, representativeness, unrecognized limits, or conservatism (Rose 1987, Tversky and Kahneman 1981, Capen 1976). Additionally, there is evidence that gender (Gustafson 1998) and even emotional state (Tixier et al. 2014) impact risk perception. Although one can attempt to minimize the effects of some of these psychological biases (Hallowell and Gambatese 2009b), opinion-based data remain severely limited in comparison to empirical data. Therefore, the needs to leverage objective raw empirical data are pressing.

### ***Level of detail of the units of analysis***

Construction work is very complex from both technological and organizational perspectives. Even though the multifactorial nature of safety risk is well known (Hallowell et al. 2011, Sacks et al. 2009), most studies have decomposed construction processes into smaller parts for the sake of simplicity (Lingard 2013). Such breakdown allows researchers to model safety for a variety of units of analysis. For example, Hallowell and Gambatese (2009a) focused on specific worker motions and activities needed for formwork construction, Navon and Kolton (2006) analyzed interactions among planned tasks at height, and Huang and Hinze (2003) modeled task, location, time, human error, and age as risk factors. Trades have most commonly been adopted as the granularity level (Baradan and Usmen 2006, Jannadi and Almishari 2003, Everett 1999). A limitation of these segmented approaches that consider elements in isolation is that there are a virtually infinite number of units of analysis that must be taken into account in order to comprehensively capture safety. This has prevented the adoption of a robust, standardized way of approaching safety analysis in construction.

### ***Attribute-based approach to construction safety analysis***

The attribute-based framework for construction safety was introduced by Esmaeili and Hallowell (2012, 2011b) and Esmaeili (2012) in an effort to jointly address the data subjectivity and study segmentation limitations previously described. Indeed, this unified approach allows the extraction of standardized safety information from objective, raw textual data such as injury reports (Esmaeili et al. 2015a). Fundamental attributes are universal, context-free descriptors of the jobsite. They span construction means and methods, environmental conditions, and human factors.

To illustrate, in the following report excerpt: “employee tripped on an electrical cord while exiting job trailer”, three fundamental attributes can be identified: (1) *object on the floor*, (2) *exiting/transitioning*, and (3) *job trailer*. While simple, this approach is powerful, as any incident can be viewed as the resulting

outcome of the joint occurrence of some fundamental attributes and the presence of a worker. It follows that the same standard safety information can be extracted for any construction situation regardless of the trade, task, industry sector or part of the world in which the accident occurred.

Esmaeili and Hallowell (2012, 2011b) initially proposed short lists of fundamental attributes (14 and 34, respectively) identified from analyzing 105 fall and 300 struck-by high severity injury cases drawn from national databases. Prades Villanova (2014) and Desvignes (2014) refined and broadened these drafts to a final, robust list of 80 carefully engineered and validated attributes by manually analyzing a larger database of 2,201 injury reports featuring all injury types and severity levels. These precursors are summarized in Table 1.

However, while the attribute-based framework is particularly well-suited for leveraging big textual safety-related data, the high cost and numerous limitations of manual content analysis remained as serious obstacles to its large-scale implementation. To solve this problem, Tixier et al. (2016a) developed a NLP tool that can automatically extract the 80 attributes presented in Table 1 and various safety outcomes with high accuracy (96% in F1 score). In this study, for illustration purposes (proof of concept), we apply our methodology on an attribute data set extracted from a pool of 5,298 raw injury reports by the aforementioned NLP tool.

### **Modeling and managing safety in BIM**

Among many characterizations, we refer to Building Information Modeling (BIM) as an information-rich design technology that can be used to generate a virtual model of an infrastructure. The strength of the BIM technology stems from its ability to augment the 3D representation of a facility with a plethora of information such as schedule, specifications, and cost. It has been shown that BIM helps improve design, management, and construction operations and is beneficial for all stakeholders during the entire construction process (Kaner et al. 2008, Goedert and Meadati 2008).

Numerous efforts have focused on the integration of safety in BIM. For instance, BIM was combined with augmented reality to improve safety recommendation understanding (Lin et al. 2014, Yeh et al. 2012), and with opinion-based risk information to assist safety management for scaffolding (Collins et al. 2014). Hammad et al. (2012) proposed a method to automatically detect risks of falls and dynamically add fences, and laser scanning technology enabled missing safety components such as guardrails or nets to be flagged by comparing virtual designs to actual structures (Ciribini et al. 2011). BIM has also been paired with tracking technologies like the GPS to send alerts to workers when they enter predefined hazardous zones (Costin et al. 2014, Fullerton et al. 2009, Chae and Yoshida 2008).

In the industry, there is preliminary evidence from an active Construction Industry Institute (CII) research team that advanced work packaging (AWP) maturity correlates with safety performance (Ponticelli et al. 2015). A possible explanation lies in that AWP goes beyond a virtual BIM model to describe not only the model component that gets built but also how it gets built in terms of specific, quantifiable work tasks. The latter is particularly well suited to safety clash detection because the work task granularity of work packages directly relates to describing those construction attributes pertinent to safety, significantly more than what would be indicated by a single BIM component.

Yet, no study has leveraged empirical data and produced results that can be used in BIM to identify what features of work are dangerous, when, where, and why. The present study is a first step in that direction. Here, we focus on BIM and AWP as candidate technologies because they presently pose greatest potential for implementation of our methodology and results. Actual implementation potential is extensive and will continue to broaden as technologies are introduced and mature.

### **Point of departure**

In this paper, we are interested in testing the extent to which data mining can be used to extract valuable new safety knowledge from large attribute data sets, in the form of safety-critical combinations

of attributes, or “safety clashes”. To this end, we compare two complementary state-of-the art unsupervised machine learning (ML) families of techniques, graph mining and hierarchical clustering on principal components (HCPC), on an attribute data set obtained from scanning 5,298 unstructured injury reports with Tixier et al.’s (2016a) NLP tool.

We define “construction safety clashes” as *incompatibilities among fundamental attributes of the work environment that contribute to construction injuries*. In this definition, we consider clashes to be situations where a group of attributes produce greater risk than simply the “sum of their parts.” In these situations the attribute combinations magnify risk and, in some case, pose new threats. A simplistic example of a safety clash is confined workspace and small particle, which is considered a clash because the aggregate of the two attributes poses a greater threat than the two attributes in isolation. While very useful for live onsite safety management, such information, based on binary input variables, is also ideally suited to be integrated with new technologies like BIM to proactively flag and address safety critical situations, thereby aiding prevention through design and the release of safer work packages. While all safety clashes are obviously of interest and would need to be accounted for in any BIM-based solution, in this exploratory study we are mostly interested in discovering safety clashes that are not already well-known and that would not clearly emerge based on the experience of any one person alone.

Esmaeili and Hallowell (2011b) represented the co-occurrence among fundamental attributes as networks. More precisely, they investigated hazardous connections in 105 fatal fall reports from the National Institute for Occupational Safety and Health (NIOSH) Fatality Assessment and Control Evaluation (FACE) database. In addition to the limitations inherent to the small size and nature of the data used, their analysis stayed at a basic level. For instance, no attempt was made at detecting communities in graphs. In this study, we go a step further in the sophistication of the analyses and in the size, diversity, and relevance of the data used.

Also, some similarities are shared by our work and that of Palamara et al. (2011), who used another data mining technique, self-organizing maps, to analyze a database of 1,207 accident reports from the Italian wood manufacturing industry. In addition to the notable differences in the data used, scope, and methodology, the information available for each report in the national database studied by Palamara et al. (2011) had been pre-filled for four categories (activity, deviation, contact and material, and mixed activity descriptors). This classification scheme fundamentally differs from the attribute-based framework we use in this study.

Finally, the entire approach of Esmaeili and Hallowell (2011b) is based on the assumption that only frequent associations of attributes should be considered dangerous, and Palamara et al. (2011) aimed at uncovering the most frequent sequences of events leading to accidents. Our effort differs from these previous studies as we assume that valuable new safety knowledge may also be found in infrequent attribute combinations.

## ANALYSIS

### Presentation of the data set

A data set of 5,298 injury reports featuring all types of injuries was obtained from more than 470 private construction organizations involved in industrial, energy, infrastructure, and mining work. The reader is encouraged to refer to Prades Villanova (2014) and Desvignes (2014) for more information about these data. The unstructured, naturally occurring reports were automatically scanned for the 80 attributes shown in Table 1 by Tixier et al. (2016a)'s NLP system. Of the 5,298 reports, 911 were not associated with any attribute and were removed, making for a final data set  $X$  of  $r = 4,387$  reports by  $p = 80$  attributes presented in Figure 1. The entries  $X_{r,p}$  of  $X$  take on the value “1” if the  $p^{\text{th}}$  attribute has been detected in the  $r^{\text{th}}$  injury report, and “0” else. The attribute counts in this final data set are reported in Table 1.

$$X = \left[ \begin{array}{ccccccc} 0 & 1 & 0 & \cdots & 1 & 0 & 1 \\ 0 & 0 & 1 & \cdots & 1 & 0 & 0 \\ 1 & 0 & 0 & \cdots & 0 & 0 & 1 \\ 0 & 1 & 1 & \cdots & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 1 & 0 & \cdots & 0 & 1 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 1 \\ 1 & 1 & 0 & \cdots & 1 & 0 & 0 \end{array} \right] \quad \begin{array}{l} p = 80 \text{ binary attributes} \\ r = 4387 \text{ injury reports} \end{array}$$

**Figure 1. Structure of the attribute data set used in this study**

As one can see in Table 1, attributes are classified in three categories: *upstream*, *transitional*, and *downstream*. Upstream precursors can be anticipated as soon as during the design phase, transitional precursors can be detected before construction begins based on knowledge of construction means and methods, and downstream precursors can only be observed during the construction phase. Note that this classification scheme may be changed and does not incur any loss of generality in the subsequent analyses.

### Overview of the methods employed

Our proposed methodology is based on two state-of-the-art, independent, complementary families of data analysis techniques, *graph mining* and *hierarchical clustering on principal components* (HCPC). To identify candidate safety clashes, we relied on community detection algorithms and edge centrality measures and on the ability of hierarchical clustering to isolate outliers into small clusters, respectively for the graph and HCPC part. In this paper a cluster refers to reports that are close to each other in the highly dimensional attribute space.

In every case, it is important to understand that the role of the algorithms is to facilitate the job of the user. The goal here is to discover new safety knowledge by isolating a small amount of highly relevant

atypical observations (ideally, a few dozens of reports) from the bulk of the data (tens or hundreds of thousands of reports in practice). Although the outcomes of quantitative methods can provide evidence for unusual or unexpected associations, human inspection and qualitative analysis is needed to decide which information is relevant. For example, not all atypical observations may be considered legitimate safety clashes, and not all safety clashes may be of interest. Logical tests that accompany quantitative results are essential. In what follows, for each approach, (1) a theoretical background is provided, (2) the proposed methodology is described, and (3) some results are presented for illustration purposes.

**Table 1. Attribute counts in our data set**

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

\*PPE = Personal Protective Equipment

## **Graph Mining**

### ***Overview and definition***

A graph, or network, is defined as a set of vertices and a set of edges (Schaeffer 2007). Vertices, or nodes, represent variables, and edges represent links between them. Mining graphs is a very effective way of identifying the key players and better understanding the interplay among a set of variables. Indeed, graphs can capture and represent the structure of many real and abstract complex systems (Fortunato 2010, Schaeffer 2007, Newman 2006). For instance, graphs have been used to describe and analyze the interaction among proteins, DNA, RNA, and metabolites within cells (del Sol et al. 2010), brain organization (Bullmore and Sporns 2009), power grids (Watts and Strogatz 1998), the World Wide Web (Dorogovtsev and Mendes 2013), or disease propagation among a population (Borgatti 2005). In the engineering management field, networks have been used to model risk and people interaction during projects (e.g., Yang and Zou 2014, Fang et al. 2012, Chinowsky et al. 2009), safety communication among workers (Alsamadani et al. 2013), and fall hazards on construction sites (Esmaeili and Hallowell 2011b).

### ***Representing attribute data sets as graphs***

We create undirected graphs where each node is an attribute and there is an edge between two nodes if the two attributes they represent co-occur in at least one injury report. Furthermore, edges are weighted according to the number of co-occurrences counts.

### ***Centrality metrics***

There are many ways to define the importance, or centrality, of a given vertex or edge in a given network. We used three of the most standard centrality measures used in graph theory, and briefly present them in what follows. Note that while conceptually related, these metrics were showed to capture and reflect different aspects of network centrality (Valente et al. 2008). For brevity and because the metrics are widely applied and mathematically simple we do not provide equations and detailed interpretations. For

further information on node eigenvector centrality, node closeness, and edge betweenness, we refer the reader to Borgatti (2005), Freeman (1979), and Girvan and Newman (2002), respectively.

*Node eigenvector centrality.* This metric takes into account the number of direct connections of a given node (known as its degree, Borgatti 2005) but also the degrees of its connections themselves. In other words, it measures the importance of a node by taking into account both the quantity and the quality of its contacts. Unlike with degree centrality, a node with only a few neighbors can be considered important in terms of eigenvector centrality if its neighbors are central (Ruhnau 2000, Bonacich 1972).

*Node closeness.* A vertex is central in terms of *closeness* if it is located at short distances from all the other nodes in the graph (Freeman 1979). In the social domain, an individual scoring high for closeness is one that would be able to communicate with all the other persons in the network at minimum time and cost, and by utilizing very few intermediaries. Therefore, a major difference with eigenvector centrality is that closeness is related to the notion of *independence*. While a vertex highly central in terms of eigenvector relies on its connections to spread its influence throughout the network, a node high on closeness can pervade the network by itself.

*Edge betweenness.* This measure is defined as the number of shortest paths that pass through a given edge (Girvan and Newman 2002). Edges with high betweenness usually act as bridges between communities, connecting the members of one group to those of another. They act as network flow controllers and coordinators, as they have the power to pass on or to retain information (Freeman 1979).

The top nodes for degree, eigenvector centrality, and closeness, and the top edge for edge betweenness, are illustrated for a simple network in Figure 2.

### ***Community structure***

While computing centrality metrics is an obvious first step in analyzing a graph, detecting communities is also very insightful. Even though there is no unique formal definition, communities (or clusters) of a graph are typically considered to be groups of nodes within which connections are dense, and between which they are sparse (Newman and Girvan 2004). For instance, two groups clearly emerge in the simple graph shown in Figure 2c (notice the two shaded areas). Many natural and human-produced networks exhibit community structure (Newman 2006). Interestingly, nodes belonging to the same communities often share unique properties and perform specific functions (Karrer et al. 2007). For instance, proteins belonging to the same clusters within metabolic networks were found to have the same role and be involved in the same cell processes (Jonsson et al. 2006). Furthermore, nodes lying at the center of their communities usually play a role of control and stability, while the ones located at the boundary often act as mediators and flow controllers (Fortunato 2010). For all these reasons, community detection is a central task of graph mining (Fortunato 2010, Lancichinetti and Fortunato 2010, Blondel et al. 2008, Schaeffer 2007). In this study, we used several community detection algorithms (presented in what follows) to identify groups of frequently co-occurring attributes but also not already well-known associations between attributes. More precisely, we assumed that bridges between communities would make good safety clash candidates.

### ***Community detection algorithms***

Following recommendations in the literature (Lancichinetti and Fortunato 2010), we used an ensemble of five state-of-the-art community detection algorithms to ensure that the clusters we found were significant and robust. We implemented these algorithms in R via the “igraph” package (Csardi and Nepusz 2006). They are briefly presented in what follows. Modularity is a widely used function in graph analysis that measures the strength of a given partition of a network into groups, by comparing the number of within-

community edges in the clustered network to the expected such number in a null model (Newman and Girvan 2004).

*Fast greedy.* The fast greedy algorithm (Clauset et al. 2004) starts with all vertices as a cluster of their own, and repeatedly merges the pair of clusters whose combination produces the largest modularity gain. This process is repeated until a single community remains. The best partition is finally selected among all possibilities as the one associated with the greatest value in modularity.

*Multi-level.* With the multi-level algorithm (Blondel et al. 2008), all nodes start as a community of their own too. At each iteration, nodes are first moved to the community of their neighbors such that the greatest gain in modularity is achieved. Second, the communities found are turned into nodes, yielding a new graph, and the first step is repeated. This process iterates until a maximum in modularity is reached and no more change occurs.

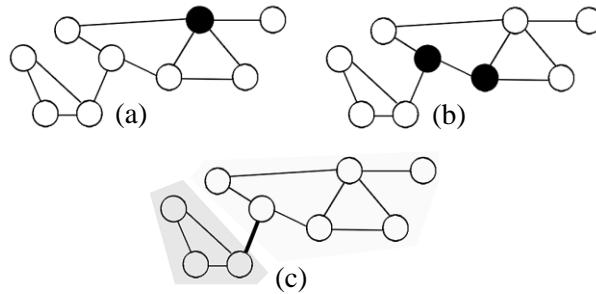
*Leading eigenvector.* The leading eigenvector algorithm (Newman 2006) recursively partitions communities (initially the entire graph) into two groups according to the signs of the elements of the leading eigenvector of the modularity matrix, and stops when all communities are indivisible.

*Spinglass.* The spinglass algorithm (Reichardt and Bornholdt 2006) is an approach based on statistical mechanics. It maps the graph to a Potts-like system with nearest-neighbors interaction where nodes are at first randomly assigned a spin state. It then uses a global optimizer, simulated annealing, to find the configuration of the system that minimizes the total energy. This ground state corresponds to the best partition of the graph into communities, which are defined as clusters of nodes sharing the same spin alignment.

*Walktrap.* Finally, the walktrap algorithm (Pons and Latapy 2005) uses agglomerative hierarchical clustering with a distance based on random walks. The assumption is that random walks tend to get

trapped into dense portions of the graph (i.e., communities). We followed Pons and Latapy (2005)'s advice and used random walks of length 2, since our graphs were quite dense. For all the other algorithms, we stuck to the default parameter values.

For each graph analyzed in this study, we implemented the five community detection algorithms presented above once with the exception of the *Springlass*, which was implemented 100 times (with majority vote aggregation) as it is stochastic. The majority vote of the 5 algorithms was used as the final community structure. When consensus could not be reached for a given node (i.e., 2 votes against 2 votes), the decision was left to the algorithm yielding the best partition in terms of modularity.



**Figure 2. Top nodes for (a) eigenvector centrality, (b) closeness, and (c) top edge for edge betweenness with two natural communities**

#### *Construction accidents as attribute network perturbations*

A graph perturbation is any topological modification of a network such as the deletion or addition of a node or edge. In genetics, perturbations in gene regulatory networks have been discovered to be one of the root causes of certain diseases. We posit that in the same way, perturbations in networks of fundamental construction attributes are one of the root causes of injuries. In what follows, this analogy is elaborated.

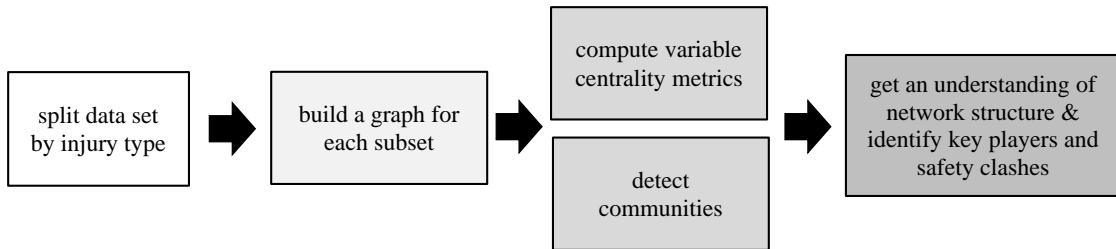
Functional states of cells correspond to stable states of an underlying gene regulatory network (Huang et al. 2009). The robustness of such networks against perturbations allows cells to constantly adapt and

continue to function normally when faced with changing conditions, such as changes in temperature and pH, or exposure to DNA-damaging agents (del Sol et al. 2010). Interestingly, it has been shown that while these regulatory networks exhibit robustness to most attacks, they are very fragile to specific perturbations, such as the mutation of one single gene or the exposure to particular toxins (Stelling et al. 2004). When faced with these perturbations, gene regulatory networks can transition into pre-existing pathological states, leading to cascading failures and to the development of diseases (Huang et al. 2009). For instance, Taylor et al. (2009) showed that topological transformations in the protein-protein interaction networks of sick patients directly impact disease outcome. Especially, specific hub proteins, critical to networks' connectivity, were frequently found to have mutated among negative outcome patients, effectively altering organization and flow of the protein network. Taylor et al. (2009) concluded that these hubs should become therapy targets.

Similarly, jobsite conditions at any particular location and at any given point in time can be represented by a combination of fundamental construction attributes, that is, by a given attribute network. Thanks to injury prevention techniques such as safety rules and guidelines, preventive and corrective measures, site supervision, and worker vigilance, networks of attributes tend to stay in stable states most of the time, despite the numerous perturbations caused by the dynamic and ever-evolving nature of construction environments. Because of this inherent resilience, a dormant hazardous situation may go unnoticed or ignored for a long time. It is indeed well known that construction workers can expose themselves to unsuspected risk because they fail to recognize latent hazards in their environment (Albert et al. 2014, Carter and Smith 2006). We postulate that perturbations in attribute networks can trigger the transition from inactive hazardous states to active accident-prone states. Under these conditions, if perturbations go unnoticed, and if no corrective action is taken, the chances of observing injuries are greatly increased. To validate this theory however, comparing graphs of injury reports to graphs of “non-injury” cases would be necessary.

According to the *construction accident as attribute network perturbations* theory, the goal of safety management would consist in ensuring that attribute networks always stay in stable states. Therefore, in the exact same way that specific DNA-binding proteins (del Sol et al. 2010) or hub proteins (Taylor et al. 2009) have become drug targets, safety-critical topological features of graphs of attributes (e.g., “safety clashes”) should become the targets of safety intervention programs and the center of attention when developing preventive strategies.

### Graph mining: Results



**Figure 3: Overall graph mining procedure**

As shown in Figure 3, the initial data set was split based on the safety outcome *injury type* in order to ease interpretation of the results. This gave 5 smaller data sets: struck by or against (2,389 reports), caught in or compressed (350), fall on same or to lower level (570), overexertion (567), and exposure to harmful substance (525). Furthermore, for each subset, the attributes that appeared in less than 1% of the reports were removed as a cleanup pre-processing step. The graph mining steps previously described were then applied. The results are shown in Table 2, and Figures 3 to 7.

We used the Fruchterman-Reingold (1991) force-directed layout algorithm available in the “igraph” R package (Csardi and Nepusz 2006) to plot the graphs presented in Figures 3 to 7. Note that the nodes belonging to the same communities are grouped together for convenience. In Figures 3 to 7, the top five attributes for eigenvector centrality are colored in dark grey, the top five attributes in terms of closeness

are filled in light grey, and the top five edges for edge betweenness are shown in red. For a given graph, less than ten nodes can be colored in grey when there is some overlap between the top attributes for closeness and the top ones for betweenness. The transparency and width of the edges is proportional to the strength of the co-occurrence between the vertices they link. In other words, attributes frequently found together in injury reports are linked by dark, thick edges, while attributes that only seldom jointly appear are connected by light, thin edges. The top 5 attributes for eigenvector centrality and closeness are also reported in Table 2 for each graph, with the top five edges for betweenness.

### ***Interpretation of the results***

In this section, we jointly interpret the graphs shown in Figures 3 to 7. For each graph, we highlight relevant candidate safety clashes and provide corresponding anonymized report excerpts for illustration purposes. A selection of clashes are summarized in Table 2 for each graph, along with the top nodes and edges for each centrality metric.

We tried to identify safety clashes by searching notable structural elements of graphs, such as edges scoring high for betweenness (shown in red in the graphs), interesting links between two or more attributes (safety-critical “chains”), or bridges between communities. As previously explained, our assumption was that interesting, not already well-known safety clashes would most likely be found among less frequent attribute combinations. This is why we did not limit our search to only the hubs or the thicker edges. Note that most of the time, the top edges for betweenness were found among bridges between communities, which is in accordance with Girvan and Newman (2002).

For the “struck by or against” graph (see Figure 3), the attributes *welding*, *grinding*, *chipping*, *slag*, *tank*, *confined workspace*, *wind*, and *working overhead*, are all grouped around *small particle* in the same community (in the upper left corner). The implication is that workers are frequently subjected to small

particle related injuries when grinding, chipping, welding, or cleaning, especially in enclosed spaces, when working overhead, or when working outside in windy conditions.

*Employee received a foreign body in the eye while cutting overhead.*

The fact that *small particle* scores high for eigenvector centrality (quantity but also importance of the connections) confirms its central position in the community and indicates that it contributed to large amounts of struck by injuries in the data set we analyzed.

Similarly, *hammer*, *nail*, *lumber*, *formwork*, *concrete*, *stripping*, *rebar*, and *wire* are clustered into the same community (in the bottom left corner of Figure 3). Note that *hammer* holds a central place within this community, making this attribute very specific and representative of its group, while some other attributes such as *concrete* or *lumber* lie on the periphery of the community, highlighting their intermediary positions with other groups (respectively the *small particle* and the *scaffold* group).

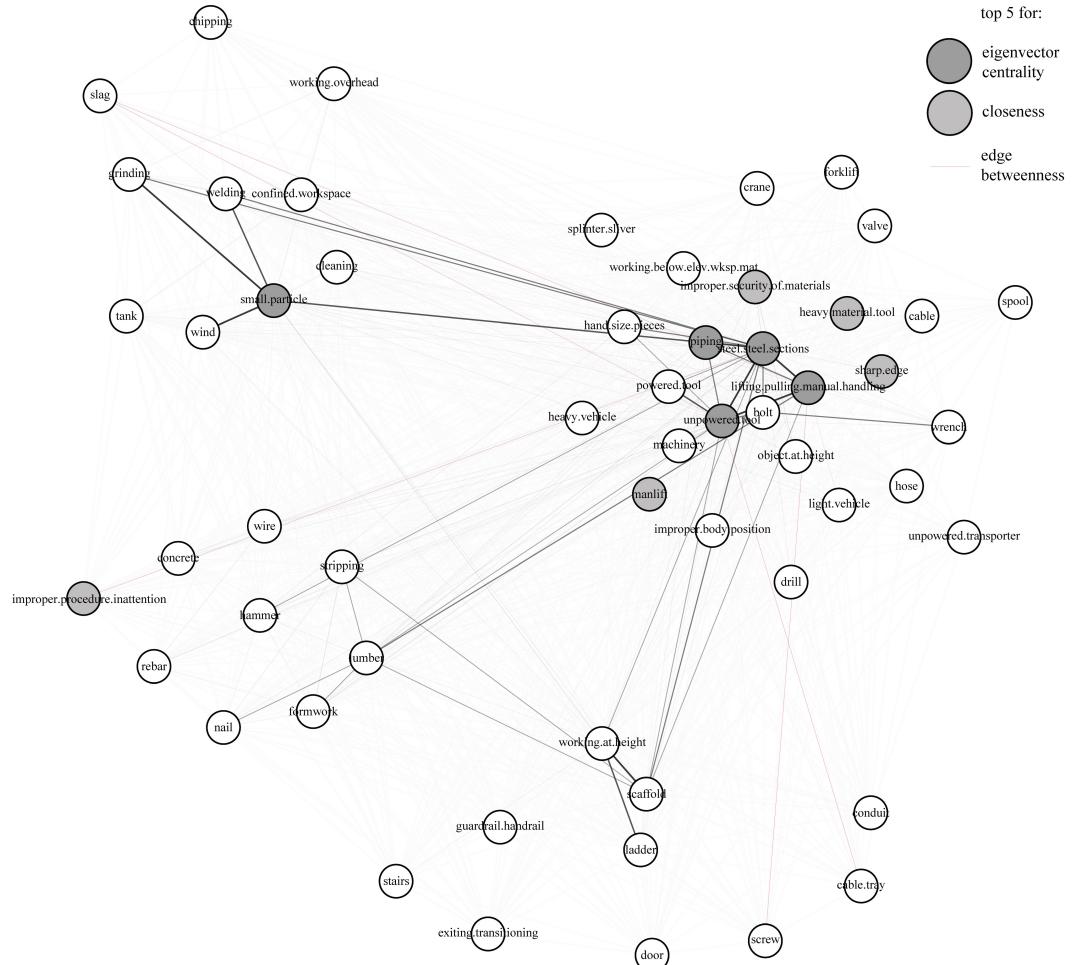
Interestingly, the attribute *improper procedure/inattention* is found in the same group, which tends to indicate that many hammer-related “struck by” injuries are caused by misses:

*While using a small crow bar and hammer to remove rust, the carpenter missed the crow bar striking his left index finger with the hammer.*

What also makes sense is that *improper procedure/inattention* acts as a bridge with other groups (especially, notice the strong connection with the large community in the upper right corner). Indeed, human error is not specific to a particular suite of actions or work situations and can be found everywhere. It is thus understandable that attributes purely and simply related to human error are shared across all communities.

Still for the “struck by or against” graph, one should note that 2 out of the 5 top attributes for closeness are related to human behavior (*improper security of tools* and *improper procedure/inattention*). Even more interesting is that this phenomenon is not observed for the other graphs. Recall that nodes of a graph scoring high on closeness are located at short distances from all the nodes in the graph. In other words, the attributes they represent are pervasive in the underlying data set. Therefore, one interpretation is that human error is prevalent, omnipresent in the “struck by or against” injury cases.

Surprisingly, *manlift* is one of the top scoring vertices on closeness, meaning that it is located at very short distance from all the other attributes in the network. In other words, it caused injuries in association with a large variety of attributes. This can be explained by the fact that manlifts are found in many different construction situations involving their direct use or not.



**Figure 4:** attribute co-occurrence graph for *struck by or against* (2,389 reports)

Regarding the “fall on same or to lower level” graph shown in Figure 4, the connection between *machinery* and *exiting/transitioning* is worth noticing and is a good illustration of a non-trivial safety clash. Excerpts of some corresponding injury reports are provided below:

*Employee was climbing out of the excavator and rolled his ankle in the process.*

*Worker was walking out of a trailer. When her foot touched the ground, she rolled her left ankle.*

Note that *exiting/transitioning* is also found in the same community as *manlift*, *light vehicle* and *ladder*, which indicates that descending or ascending is problematic not only for *machinery* but more generally for any kind of equipment.

Furthermore, the strong link between *object on the floor* and *unpowered tool* reveals that numerous falls on same level (tripping, stumbling) are due to the presence of loose tools on the floor, and the close proximity of these two attributes respectively with *confined workspace*, *scaffold*, and *working at height* suggests that the risks of falls due to the presence of objects on the floor is compounded in constrained spaces. Finally, from the thick edge between *object on the floor* and *piping*, it is possible to infer that many of the problematic objects left on the floor are pieces of pipes (at least in our data set):

*Employee tripped over a pipe support that was lying on the scaffold causing him to strain his knee.*

*A contractor employee stepped on a pipe that was installed just above the floor of a scaffold, causing him to roll his ankle.*

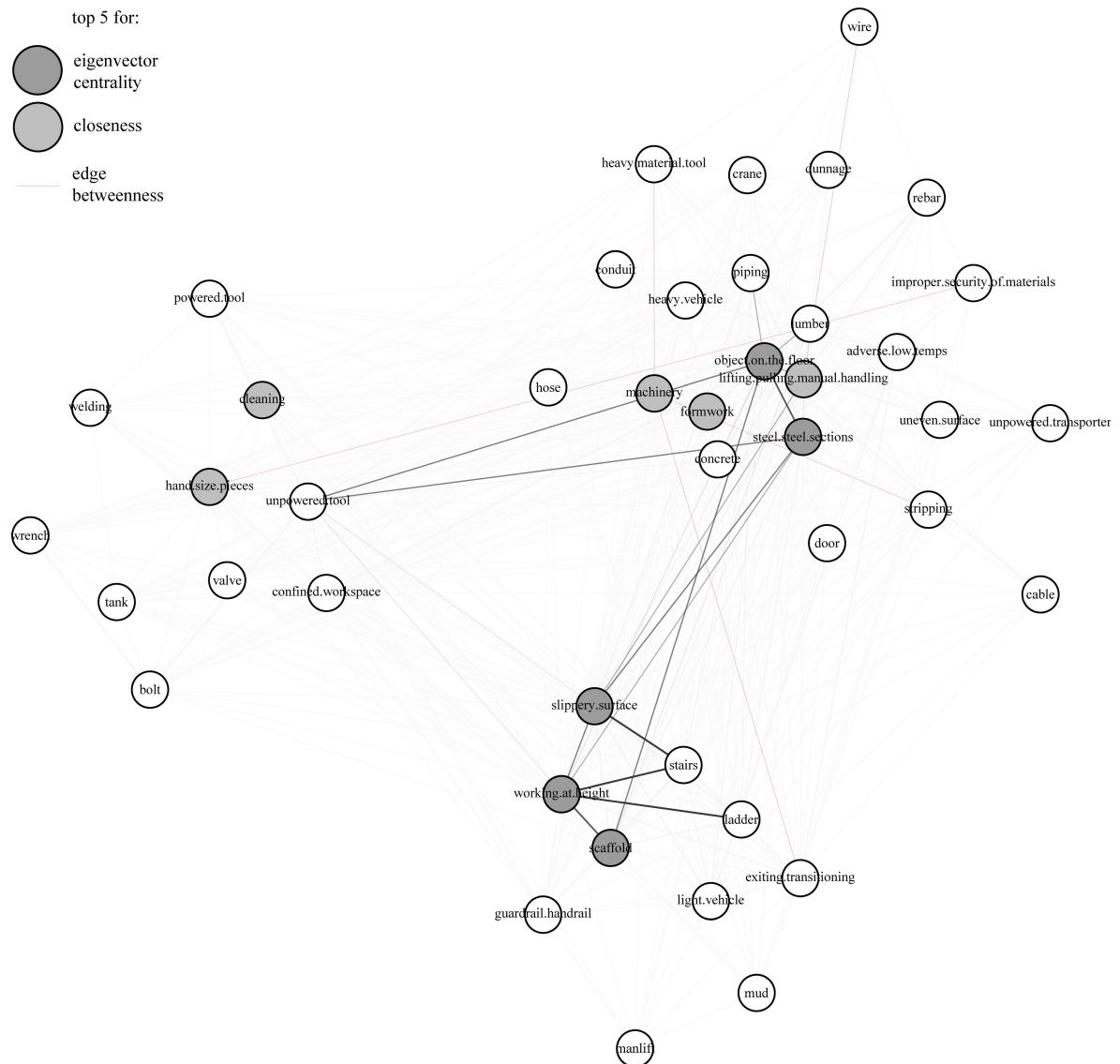
*Employee was working on a scaffold installing pipe supports, as he was working he tripped over a pipe support that was lying on the scaffold causing him to strain his knee.*

Still with respect to the “fall on same or to lower level” graph, an interesting clash is *steel/steel sections* and *slippery surface*. This clash may not appear evident or identifiable from common sense only since steel sections are not supposed to be used as walking surfaces:

*Employee was unhooking slings from metal beams inside dumpster when he slipped on slick metal from weather and twisted his right knee.*

*While traveling on foot through the waste caustics area, a subcontractor employee slipped and twisted his right knee on a steel plate that covered a U-drain in the unit.*

Note that both *steel/steel sections* and *slippery surface* are among the top 5 vertices for eigenvector centrality (many connections that are themselves central), meaning that they are major injury contributors when associated not only together but also with other attributes.



**Figure 5:** attribute co-occurrence graph for *fall on same or to lower level* (350 reports)

Quite logically in the “overexertion” graph shown in Figure 5, a major clash is the strong association between *lifting/pulling/manual handling* (a highly central node in terms of eigenvector centrality), *heavy material/tool*, and *improper body position*:

*Employee complained of soreness in his elbow after handling heavy air impact guns and crusher teeth (each approx. 50 lbs). The employee was positioned on the fixed end of the crusher, which has less space to work and involves awkward positioning. Two weeks later the employee felt numbness while manipulating a 4 lbs hammer, causing them to drop the hammer.*

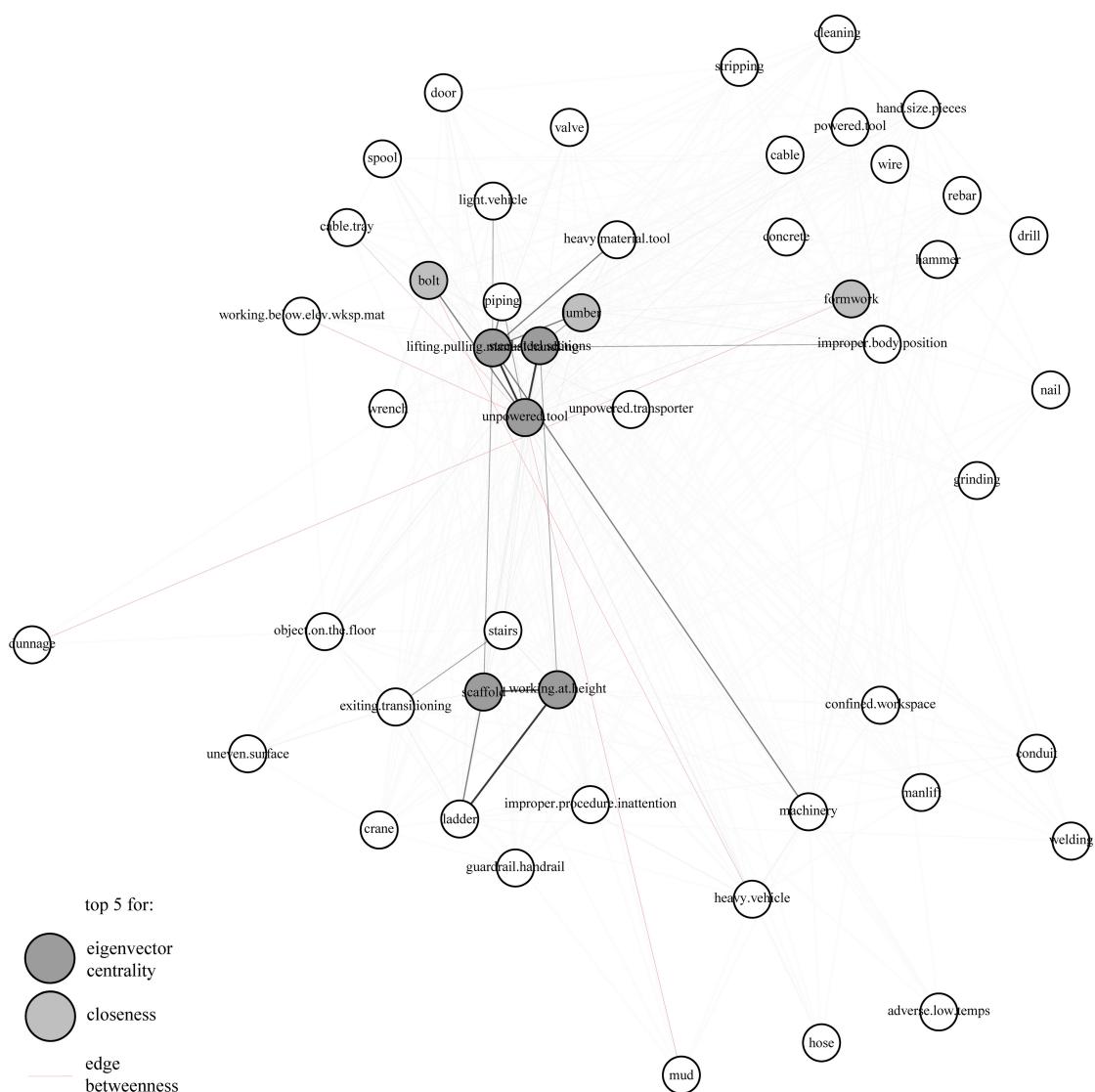
Very related to the safety clash above is the one involving *lifting/pulling/manual handling, unpowered tool* and *working below elevated workspace/material*. The attribute *working below elevated workspace/material* produces effects that include that of *improper body positioning*:

*This employee crawled under the air channel pipe to grab a chain fall and felt a discomfort in his abdomen area when he started to get up.*

But the potential adverse effects of *working below elevated workspace/material* are not limited to that of *improper body positioning*. For instance, in the cased depicted by the excerpt below, body positioning seems right. What is problematic is the relative position of the employee relative to the tool they are manipulating:

*An iron worker had to go under some pipe to connect a come-a-long to a beam. When he was under the pipe he pulled the chain towards himself. When he did, the end of the come-a-long struck the employee in the mouth.*

One should note too that the *unpowered tool - working below elevated workspace/material* edge stands in the top 5 edges for betweenness, meaning that it is in a position of controlling the flow in the network. In other words, in the “overexertion” graph, one of the fastest way to link two non-neighbor attributes is by passing through the *unpowered tool - working below elevated workspace/material* edge.



**Figure 6: attribute co-occurrence graph for *overexertion* (570 reports)**

Finally, the close proximity of *lifting/pulling/manual handling* to *bolt*, *wrench* and *unpowered tool* implies that many overexertion injuries occur when tightening bolts. This makes sense. This observation is strengthened by the fact that *unpowered tool* stands among the top 5 nodes for eigenvector centrality (that reflects both the quantity and the quality of connections) and that *bolt* is very high for closeness (a measure of pervasiveness throughout the network).

*EE was tightening bolts on non- segmented bus duct to the specified torque value when he felt discomfort in his back.*

*A subcontractor worker was adjusting bolts on a joint with a ratchet tool. He felt a strain in his neck.*

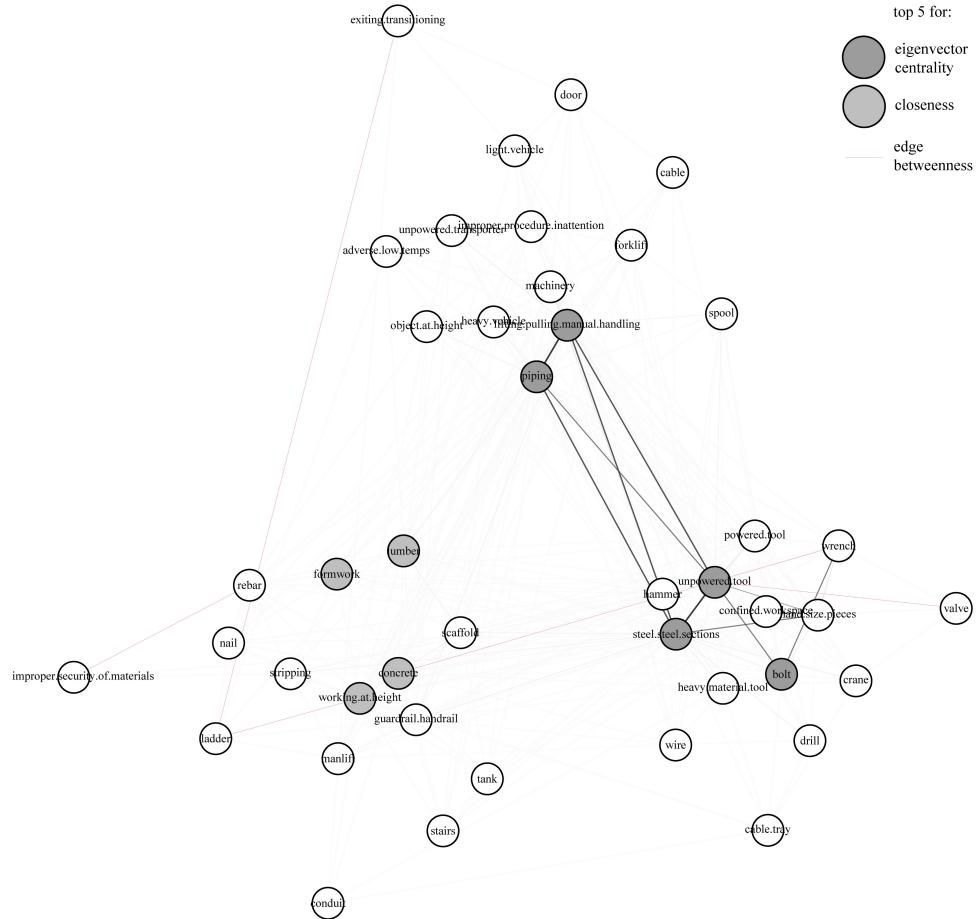
A major clash that can be identified from the “caught in or compressed” graph shown in Figure 6 is *improper security of materials and rebar*. It seems that many pinches and crushing injuries indeed involve improperly secured or unprotected rebar (either loose rebar or protruding rebar in place).

*An ironworker was placing rebar when the bar dropped, pinching his finger.*

Moreover, the close proximity of the aforementioned pair of attributes to *formwork* and *lumber*, two attributes in the top 5 for closeness (measure of pervasiveness throughout the network) indicates that improperly secured *rebar* is often found within the context of handling lumber:

*As he was removing a 16' long 2x4, the board became top heavy and pinched his finger between the board and the horizontal rebar protruding from the construction joint.*

*Employee pinched his finger onto rebar, while positioning formwork.*



**Figure 7: attribute co-occurrence graph for *caught-in or compressed* (567 reports)**

The very strong bond between *piping* and *lifting/pulling/manual handling* (two attributes in the top 5 for eigenvector centrality) reveals that this association is responsible for many “caught in or compressed” injuries. This can be readily understood. Actually, pipes are heavy, and move very easily by nature. Positioning or manipulating them is therefore prone to creating crush or pinch injuries. This is compounded by the fact that pipes are often to be installed in confined spaces (notice the strong links with the *confined workspace* community on the bottom left) where the proximity with other hard surfaces is high, and at height (suspended).

*The pipe was suspended about 2”–3” off the ground. While installing the clamp the pipe moved and slipped from his hands causing it to fall to the ground where the worker crushed his left hand index finger.*

*Piping* is also strongly connected with the community of *unpowered tool, bolt, and steel sections* (among others). These attributes are highly central in terms of eigenvector centrality, denoting that they are major “caught in or compressed” injury contributors. The strong link with *piping* suggests that all these attributes play as a team:

*Worker pinched his finger between a bolt flange and a pipe.*

*Worker pinched his hand between a pipe and a piece of steel.*

Also, the link between this community (which also includes, among others, *hammer*) and *concrete* is interesting:

*Employee was stripping floor beam recessed block out, set cats paw with hammer, missed cats paw and pinched left hand between tool and concrete.*

Finally, many “caught in or compressed” injuries involve ladder, as can be concluded from the observation that *ladder* is the endpoint of two edges in the top 5 for edge betweenness (flow controllers, bridges between groups).

*Ladder slipped out of employees hand and pinched their right middle finger*

Expectedly, *hazardous substance* holds a very central place in the graph shown in Figure 7 (“exposure to harmful substance”). Its direct connections with *concrete*, *chipping*, *grinding*, *lumber* and *small particle* shows that exposure to fine dust was a frequent issue in the data set we analyzed. Especially under windy conditions (notice the very strong link between *hazardous substance* and *wind*):

*While burning the bolt heads, the dust from the demolition of the concrete deck that had previously been demolished in the same area was blown by the wind and got in the eye of the employee.*

*As employee was working adjacent to a wood cutting operation, wind blew saw dust into his eye.*

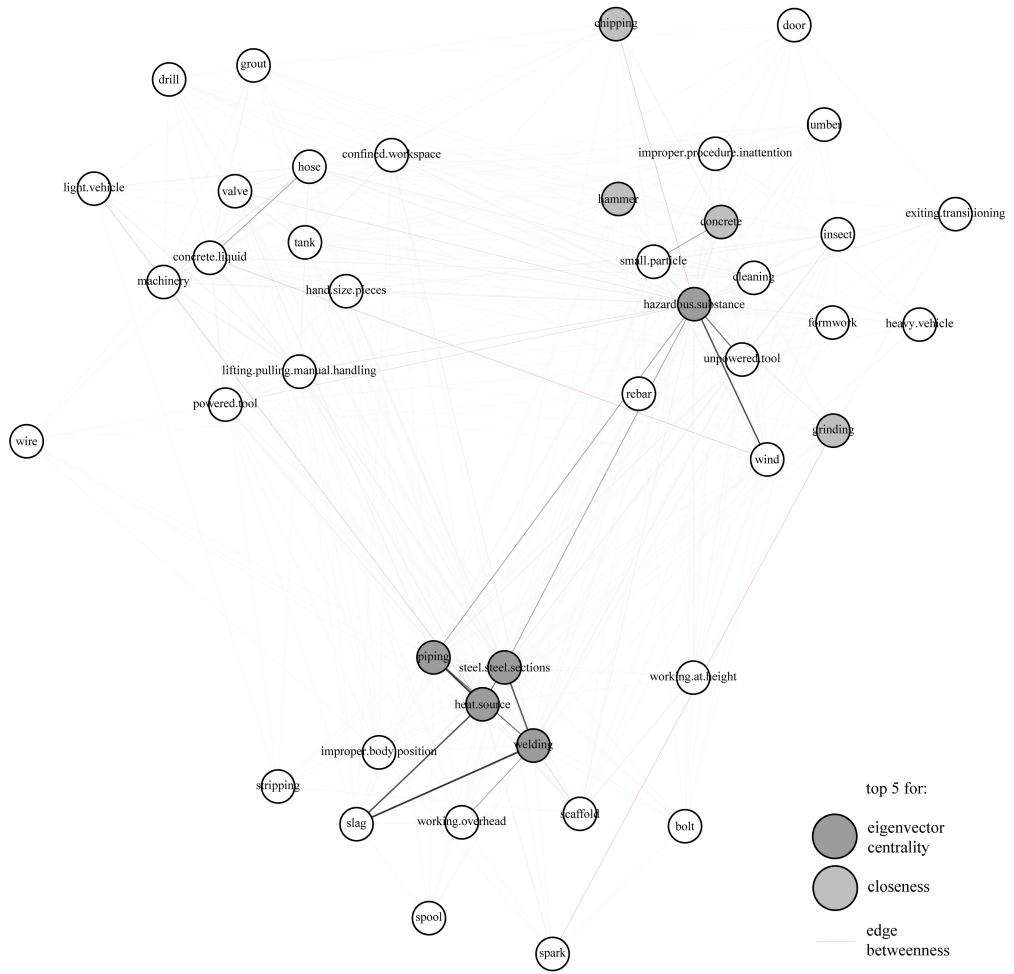
*It was a windy day and an insulator employee was cutting a piece of piping insulation. The employee felt a discomfort in his right eye.*

The grouping of *concrete liquid*, *grout*, *machinery*, *hose*, *valve* and others into the same community (in the upper left corner) is not surprising either.

*Worker noticed that the adjacent concrete was drying out and needed to be wet to maintain good cure. When worker picked up the water hose the valve opened and a blast of hot water came in contact with his abdomen causing the burn.*

Maybe more interesting in the “exposure to harmful substance” graph is the close proximity of *welding* to *improper body positioning*, *working overhead*, and *scaffold*. It is well known that *welding* (directly or through *slag* and *spark*) is a *major* light and heat sources and is thus central in creating “exposure to harmful substance” injuries. However, it appears that the risk of this attribute is compounded when workers adopt non-natural body positioning:

*Employee was welding overhead and felt slag fall on right side of neck resulting in a burn.*



**Figure 8: attribute co-occurrence graph for *exposure to harmful substance* (525 reports)**

Notice that logically, there is a direct link between *improper body positioning* and *confined workspace*, suggesting that the former could be consequence of the latter:

*Employee had a few inches between him and the weld he was making; it was a very tight and awkward position. He was exposed to arc flash that came in from under his hood*

Non-natural body positioning is also an issue with other hazardous substances such as *concrete liquid* or insulation:

*While reaching overhead, some of the patch mix that the carpenter was using got between his shirt and glove, causing minor concrete burn.*

*A Carpenter Foreman got some insulation in his eye while stuffing insulation overhead between the hard lid trusses.*

Finally, the connection between *piping* and *confined workspace* revealed an interesting, non-trivial clash:

*Employee slipped into a small excavation containing very hot water (app. 158 degrees Fahrenheit). He sustained severe burns to both legs. Water coming from the melting snow was heated by a steam line installed the previous day.*

Overall, we showed that the community structure exhibited by the attribute data makes physical sense and that graphical features can be used to identify interesting combinations of attributes. This shows the promising potential of our methodology and tends to validate Tixier et al.'s (2016a) NLP tool with which the attribute data set was extracted from injury reports in the first place.

**Table 2. Top elements for eigenvector centrality, closeness, and edge betweenness**

Injury type	eigenvector centrality	closeness	edge betweenness
Struck by or against	unpowered tool, small particle, piping, manual handling, steel sections	heavy material/tool, sharp edge, improper security of materials, manlift, improper procedure/inattention	working below elevated wksp & slag; cable tray & unpowered tool, imp. procedure/inattention & steel sections
Caught in or compressed	bolt, steel sections, unpowered tool, piping	formwork, lumber, concrete, working at height	improper security of materials & rebar; valve & unpowered tool; ladder & exiting/transiting.
Fall on same or to lower level	object on the floor, working at height, slippery walking surface, scaffold, steel sections	hand size pieces, formwork, machinery, manual handling, cleaning	machinery & exiting/transiting; scaffold & object on the floor.
Overexertion	lifting/pulling/manual handling, unpowered tool, steel sections.	formwork, bolt, lumber, scaffold, working at height	heavy material/tool & improper body positioning, spool & light vehicle, working below elevated workspace/material & unpowered tool.
Exposure to harmful substance	hazardous substance, piping, welding, heat source, steel sections	concrete, unpowered tool, cleaning, hammer	wind & hazardous substance; piping & confined workspace; concrete liquid & wind; welding & working overhead.

To mine the attribute data set from a different perspective and gather complementary safety knowledge, we used hierarchical clustering, as shown next.

## Hierarchical Clustering on Principal Components (HCPC)

### *Overview and definition*

To identify atypical but valid combinations of attributes from another perspective than a strictly “social” one, we used an unsupervised data mining technique complementary to network analysis, hierarchical clustering. Hierarchical clustering is known for its ability to isolate outliers into small clusters. By manually inspecting these small clusters automatically constructed, we were able to easily identify valid cases that “stood out from the crowd”, that is, potential safety clashes.

Note that to find more stable and definite clusters and therefore enhance the robustness of our results, we used Hierarchical Clustering on Principal Components (HCPC, Husson et al. 2010), an improvement over traditional hierarchical clustering. HCPC consists of three complementary steps as shown in Figure 8: observations are (1) projected onto the principal component basis, (2) partitioned into groups via agglomerative hierarchical clustering, and finally, (3) the groupings are consolidated by using a K-means algorithm. In what follows, these three steps are detailed.



**Figure 9: Hierarchical Clustering on Principal Components (HCPC) steps**

### *Step 1: Principal Component Analysis*

First, it is necessary to recall that in this study, each observation (i.e., each injury report) lives in an 80-dimensional space (the feature space shown in Figure 1). Each fundamental construction attribute represents a dimension of this space, and each injury report is defined by how it loads on these dimensions; that is, by its coordinates (zeroes or ones) in the feature space. Each attribute is indeed either present or absent from a given injury report. This is similar to Kauffman’s (1969) view of genes as either “on” or “off”.

Principal Component Analysis (PCA) is a common algebraic procedure used to reduce the dimensionality (i.e., the number of features) of a data set while preserving most of the information originally present in it. More precisely, PCA first constructs an orthonormal basis linearly derived from the original feature space, such that the variance of the observations projected in this new basis is maximized (Shlens 2014, Jolliffe 1986). Each axis (or principal direction) of the new basis matches the direction of maximum variability of the cloud of data points, with the constraint that each successive axis is orthogonal to all the preceding ones. Mathematically, this first phase of PCA is performed by carrying out an eigenvalue decomposition of the covariance matrix  $C_{p,p} = X^T X / (r - 1)$ , where  $X_{r,p}$  is the original data matrix shown in Figure 1, centered (column means subtracted for each column). One should note that this approach is equivalent to performing a singular value decomposition of  $X_{r,p}$  (Shlens 2014).

Because the covariance matrix  $C_{p,p}$  is real and symmetric, it is orthogonally diagonalizable, that is, there is an orthogonal matrix  $V_{p,p}$  such that  $L_{p,p} = V^T C V$  is diagonal. This expression is equivalent to  $C = V L V^T$ . The matrix  $V_{p,p}$  contains as its columns the  $p$  eigenvectors (also called the principal directions, or principal axes) of the square matrix  $C_{p,p}$ . The projections of the  $r$  observations onto the principal directions, called the principal components, are given by the rows of  $X V_{r,p}$ . The columns of  $X V$  represent how each observation loads onto each dimension of the new basis (the eigenvectors given by the columns of  $V$ ). One can use this mapping to go from the original feature space to the principal component space, and vice versa. The matrix  $L_{p,p}$  contains the eigenvalues  $\lambda_1 \dots \lambda_p$  of  $C_{p,p}$  as its diagonal entries. These values represent the amount of variance accounted for by each of the  $p$  principal components. This first phase of PCA is variance conserving: the information is simply viewed from some optimal perspective, thanks to the change of basis.

The data reduction takes place in the second phase of PCA, sometimes called the compression phase (Smith 2002). This second step simply consists in ordering the principal components by decreasing

eigenvalues, and in selecting the first  $k$  ones (where  $k < p$ ). The dimensionality of the feature space is thus reduced from  $p$  to  $k$  while conserving most of the variance originally present in the data and discarding most of the noise. Therefore, considering observations in the space made of the first  $k$  principal directions rather than in the original feature space allows for a more effective and stable clustering (Husson et al. 2010).

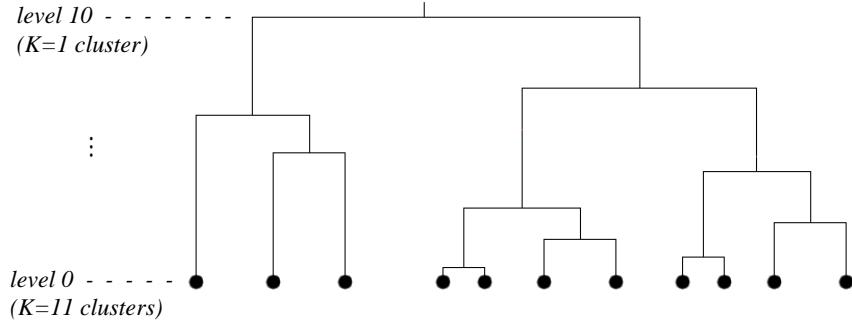
To account for the Boolean nature of attributes, we used Multiple Correspondence Analysis (Greenacre 2007, Benzécri 1973), the extension of PCA to categorical variables.

### ***Step 2: Agglomerative hierarchical clustering***

Agglomerative hierarchical clustering refers to a class of unsupervised learning algorithms that classify  $r$  observations into a hierarchy of disjoint clusters, following a recursive, bottom-up approach (Hastie et al. 2009, p. 523). As shown in Figure 9, the two clusters optimizing an objective function are combined at each step  $s$ , which results in a grouping at level  $s + 1$  with one less cluster. Initially at level 0, each data point belongs to its own cluster. The algorithm stops after  $r - 1$  steps, when all observations belong to the same group. This approach is known as Ward's method (Ward 1963). Unlike other clustering techniques like K-means or K-medoids, hierarchical clustering offers the advantage of not requiring *a priori* knowledge about the number of clusters.

We capitalized on the robustness of hierarchical clustering to outliers (Loureiro et al. 2004), in order to make atypical yet valid injury cases “stand out from the crowd”. The assumption is that because outliers are associated with unusual combinations of attributes, they will be isolated in low density areas distant from regular observations and therefore will tend to end up grouped into small clusters (Almeida et al. 2007). These outliers may correspond to errors made the NLP tool when scanning the database of injury

reports, or reveal rare but valid associations between attributes. The latter were of great interest to us as they were candidate safety clashes.



**Figure 10. Agglomerative hierarchical clustering illustration for  $r = 11$  observations**

The choice of the objective function is task-dependent. In this study, because MCA (the first step of HCPC) is variance-based, and the K-means algorithm (HCPC's last step) is based on the squared Euclidean distance, the within-cluster variance had to be used as the objective function (Husson et al. 2010). This function, defined in equation 1, measures the extent to which members of a cluster are close to the center of this cluster (James et al. 2013, p. 387). One should note that even though the injury reports have binary coordinates in the original space (the attribute space), their projections onto the principal direction basis are real numbers, as was explained in the previous section. Therefore, using the Euclidean distance was not a problem. To summarize, the two clusters minimizing the increase in within-cluster variance when merged were grouped at each step.

$$W(C_k) = \sum_{x_i \in C_k} \|x_i - \bar{x}_k\|^2$$

**Equation 1. Within-cluster variance of cluster  $C_k$ .**

Where  $\bar{x}_k$  is the mean of  $C_k$ , defined as the coordinate-wise averages of the observations in  $C_k$ , and  $\{x_1, \dots, x_r\}$  is the set of  $r$  observations. Each observation is a vector (one coordinate per dimension).

Each level of the hierarchy corresponded to a theoretically valid partition of the observations into K clusters. When hierarchical clustering is used for its primary purpose (i.e., grouping observations into a few compact and well separated clusters), the selection of the optimal level  $s_{\text{optimal}}$  of the hierarchy can be achieved quite easily based on quantitative criteria. For instance, an approach consists in selecting the level associated with the number K of clusters such that the total within-cluster variance  $W = \sum_{k=1}^K W(C_k)$  is minimal. However, when the task of interest is that of outlier detection like in this study, this criterion loses relevance, and finding the optimal level becomes more problematic. Indeed, selecting a level that is too high in the hierarchy returns only a few very big clusters, where outliers are mixed with regular observations, while picking a level that is too low yields a vast amount of very small clusters, which is not a significant improvement over parsing and analyzing reports manually. We followed Loureiro et al.'s (2004) rule of thumb to select the level of the hierarchy that returns  $\max(2, r/10)$  clusters, where r is the total number of observations (reports in our case). This heuristic still gave too many clusters (e.g., 230 clusters with the *struck by or against* data set), and therefore, we used  $\max(2, r/50)$  instead. Furthermore, of the  $\max(2, r/50)$  clusters returned, only the ones containing less than 10 observations were examined.

### ***Step 3: consolidation of the clustering via K-means***

The partition obtained in Step 2 was refined by applying a K-means algorithm in order to increase the consistency and separation of the clusters (Husson et al. 2010). The K-means algorithm consists of the following three steps (James et al. 2013, p. 388; Halkidi et al. 2001):

- i. select initial cluster centers,
- ii. for each cluster center, create a cluster by selecting the observations that are closer to this center than to any other center (in terms of Euclidean distance),

- iii. update the cluster centers (computed as the coordinate-wise means of the observations in each cluster).

Overall, the K-means algorithm seeks to minimize the total within cluster variance  $W = \sum_{k=1}^K W(C_k)$ , where  $W(C_k)$  is the within cluster variance of a given cluster  $C_k$  as defined in equation 1. Steps ii. and iii. are repeated until convergence is attained, that is, until the assignments do not change. The initial cluster centers were given by the coordinate-wise averages of the clusters found by hierarchical clustering at the previous step.

### **Hierarchical clustering on principal components: results**

Using the rule of thumb from Loureiro et al. (2004), 90 clusters were requested. 40 of them contained less than 10 elements and were manually inspected for safety clashes. Relevant findings are organized by main themes in what follows. Again, for brevity, only a few anonymized, representative reports are shown for each clash.

#### ***Congested and confined workspaces compound the risk of other attributes***

The attributes *congested workspace* and *confined workspace* act as catalysts for accidents. They increase the risk of many different attributes, and therefore have the power to turn a great variety of work situations into hazardous situations. As a result, they should always be considered main safety targets, even when other seemingly more impactful attributes are present. For instance, the risk of tripping on an object on the floor, or of being struck by or against a tool or some material is greatly increased in congested and confined spaces, as illustrated by the following examples:

*Workers were moving a piece of formwork from a congested space. While doing so, one worker tripped on rebar adjacent to the walk path, causing him to fall and twist his knee.*

*Employee performing demolition operations in close proximity to another employee was struck by that employee's hammer.*

*An electrician helper was installing conduit for lighting in close proximity to a section of pipe. His left forearm came in contact with the pipe, which had a piece of metal protruding resulting in an abrasion.*

Also, in order to adapt to congested or confined spaces, workers often have to adopt improper body positioning, or to follow improper procedures:

*Employee was tightening flange by flange connections with opposing forces of two combination wrenches. He was working in a tight area, which created poor body positioning to effectively tighten the bolts. Great amount of pain in neck and back.*

*The ladder was set in front of the last set of panels with very little room around the landing on the deck of the trailer. When the laborer went to descend the ladder he went around the grab rail due to limited access in front of the ladder. One of the hooks popped out of the rub rail and caused worker to fall. He attempted to cushion his fall with his hands and was diagnosed with a fractured wrist.*

*Two employees were trying to move a pallet with a 4000 lbs valve on it into position so they could get the pallet jack under it. The area was congested and there was nowhere to rig from. They decided to use a 4x4 post to pry the broken pallet into a more suitable position. As the employee pried the pallet he felt some pain in his lower back.*

It is the work environment that should adapt to workers, not the opposite. Just because ergonomics and safety concerns may be more difficult to address in congested and confined workspaces does not mean that they should be ignored.

***Flaggers are at greater risk for slips, trips and falls due to their attention being caught by other stimuli***

This is a good example of a previously undocumented, rather counterintuitive phenomenon that seems benign but may be responsible for many lost work time injuries every year:

*While pouring concrete for sidewalk, foreman was trying to flag down a concrete truck to the pour. With his sight directed to the truck, he did not see the curb, tripped and fell.*

*While spotting crane, employee tripped on dunnage and fell into a rebar mat striking his hand on the rebar.*

Considered separately, each of these incidents could be deemed unlikely and due to bad luck. But when analyzing large numbers of injury reports collected from hundreds of construction sites throughout the world and representing hundreds of thousands of worker hours, trends begin to emerge. One can then realize that these injuries may happen more frequently than initially thought and that they may not be random but be caused by the same underlying mechanism. This insight extraction process is a necessary step towards taking corrective actions and improving safety performance.

### ***Workers are unable to recognize immediate hazards due to poor visibility***

It has been shown that many workers involuntarily put themselves at risk not necessarily because hazards are not visible, but because workers fail to recognize their presence (Carter and Smith 2006). The remediation strategies that have been proposed in the literature involve hazard recognition training primarily based on visual information (Albert et al. 2014). This assumes that all hazards are visually identifiable. Although it may be true, it should be stressed that not being able to visually detect the presence of a hazard does not necessarily mean that this hazard is absent from the work environment. Put differently, not being able to assess the full risk profile of the working environment is in itself very problematic and should be considered hazardous in its own right.

*Employee was walking to job trailer, due to recent rainfall stepped into unseen low spot that was covered with water and fell striking his thigh.*

*Worker was walking under the module when they came to a beam they had to walk under. On the other side of the beam there was a 1" hydro vent installed that was not visible to the worker. When the worker walked under the beam, he stood up and his face contacted the drain. This caused a laceration above the eye and the abrasion was closed by 6 stitches.*

*While employee was walking on the snow covered ground he slipped on a hidden patch of ice, fell back and hit his head on the ground causing a contusion and concussion.*

*Employee stepped on the bottom form plate onto an exposed nail. Exposed nail was not visible due to murky water.*

Hinze and Teizer (2011) showed that a majority of vision-related fatalities in construction involve workers being struck by moving equipment or vehicles, and are mainly due to blind spots, obstructions and extreme lighting conditions. While valuable, these findings are more related to equipment and vehicle drivers not noticing the presence of workers on the ground rather than purely to the inability of workers to recognize latent hazards due to poor visibility. On the other hand, our findings suggest that there is another class of vision-related injuries that may be for the greater part of lower severity, and do not involve equipment. These injuries are due to workers not being able to identify the presence of immediate hazards due to lack of visibility. In these cases, additional precautions should be taken, such as safety warnings highlighting the presence of the hidden hazards, or when unfeasible, delivery of general recommendations to use extra caution in specific settings.

#### ***Exiting equipment, vehicles, or work stations is safety critical***

These very anodyne actions performed very frequently every day may seem completely harmless compared to more serious, high energy hazards, such as suspended loads or moving heavy equipment, but empirical evidence suggests that they may make important injury contributors:

*Employee reported that he twisted his knee while stepping off of the last step of the crane access ladder.*

*While exiting a van an employee had their finger pinched by the door.*

*Employee exited office trailer, rolled ankle on stair platform.*

This finding is consistent with recent psychological research that showed location shifts to disrupt visual and spatial cognitive processing and to cause forgetting (Radvansky et al. 2011). In other words, transitioning from one work area to another (which includes exiting equipment and vehicles) may decrease situational awareness, alter risk perception, and therefore increase the potential for injury.

### ***Working with hazardous substance requires proper preparation and PPE, and following procedures***

Only workers who are fully aware of their environment and are in a mental and physical state of mind conducive to full concentration should be allowed to work with or in the vicinity of hazardous substances. Also, it should be emphasized that procedures should be followed exactly in case of incident, at the risk of observing very serious consequences:

*A liquid loading foreman had just completed loading an ammonia rail car. He inadvertently struck a load hose bleed valve with his foot and ammonia sprayed onto his left knee, causing chemical burn.*

*Insulator was installing fireproofing on structural steel. As he was moving around, his back area came in contact with a valve that developed a small leak of 40 percent acrylic acid. The employee bypassed and failed to use the nearest safety shower instead he went looking for his supervisor.*

Similarly, adequate PPE should be worn on tasks dealing with hazardous substances, and PPE should be carefully inspected prior to initiating work:

*A laborer was helping grout door frames when she got a small amount of dried grout into her right eye. Upon investigation it was discovered that the laborer's safety glasses did not provide a snug fit nor was the potential of falling debris identified in the JHA.*

*Worker disconnected the grout line which was still pressurized at approximately 70-80 psi. The back pressure relief caused the cement grout to go in an upward motion, covering the workers face, hard hat and safety glasses. Some of the grout went behind his glasses and into his eyes.*

The combination of the attributes *hazardous substance* and *powered tools* (illustrated in the following report examples) is an elegant example of the theory of *injuries as perturbations in networks of attributes* posited earlier. Indeed, some inherently hazardous substances are completely harmless in a static, stable state. However, the energy transferred to these substances by a powered tool such as a for instance a grinder can trigger a transition into an unstable, hazardous state. A fiberglass pipe, for instance, is inoffensive; however when being cut, volatile and inhalable fiberglass dust is produced, posing immediate and long-term safety concerns (skin, eye, lung and stomach punctual and chronic irritation).

*An employee felt discomfort in right eye when grinding paint off a structural beam. After flushing eye on-site employee was taken off-site for medical assessment and returned to work without restriction.*

*Worker was cutting fiberglass pipe with a grinder, and when he removed his full face respirator mask he felt a foreign body sensation in eye.*

*Employee was working in the 521 area, approximately 20 foot from an area where XXXX Manufacturing was performing some grinding activities on fiberglass pipe when he felt the irritation in his left eye.*

To make sure that attribute networks stay in stable states, safety management should take corrective actions. For instance, it is now common practice to equip demolition equipment with water sprinklers to suppress concrete dust. Similar proactive strategies (e.g., ventilation, vacuuming) should be adopted at the worker level, whenever powered tools are used on hazardous substances.

## **CONCLUSION**

The data mining procedures introduced in this study, used within the attribute-based framework, allowed the automated identification of good candidate safety clashes from large data sets of attributes extracted from raw injury reports. Besides the findings themselves which are limited to the data we used and mainly serve as a proof of concept, we believe that the methodology in itself is the major contribution of this paper. It shows great promise to become a standard way of extracting safety knowledge from raw textual injury reports, and will help to replace the long-standing limitations associated with opinion-based safety analyses.

Also, the theory introduced, which posits that construction accidents are induced by perturbations in underlying networks of fundamental attributes, is promising but needs additional work to be further clarified and delineated. Specifically, attribute networks of injury cases need to be compared with that of “non-accident” cases, that is, random observations of the jobsite at times when no injuries occur. Nevertheless, preliminary empirical evidence suggests that this theory may hold. To our best knowledge, it is also the first time that hierarchical clustering is used to retrieve groups of atypical injury reports and inspect them for rare and atypical associations of attributes.

Such safety knowledge, based on binary attributes, is ideally suited for integration with systems such as BIM, and can be used to support a longitudinal approach of hazard identification and safety management that supports proactive decision-making and provides information with increasing fidelity as project planning matures. In early phases, attributes could be assigned to physical elements and spaces in BIM and the system, in turn, could automatically detect and flag safety clashes. For example, a designer would be able to identify and assign upstream attributes (i.e., those identifiable in design such as *steel sections* or *crane*) and the BIM system could then provide two useful pieces of information: (1) which clashes exist at a particular time and location based on the upstream attributes alone, and (2) the transitional and downstream attributes to which the situation is vulnerable. As the design matures to construction planning and work packaging, the attributes identified in the design phase are carried forward, new attributes are identified, and the model is refined accordingly. For example, transitional attributes (i.e., those identifiable in construction planning such as *forklift* or *overhead work*) can be added during work packaging as construction means and methods are selected. Finally, as construction begins, downstream attributes (i.e., those identifiable only once work begins such as *poor housekeeping* or *poor visibility*) can be added and expected clashes can be removed or communicated to the workforce during pre-task planning meetings.

Although our discussion of the implementation of the presented methodology and results is focused on BIM and AWP, the potential applications are far more extensive. We postulate that the ability to proactively identify safety clashes can provide useful information in any data-driven technology and many safety planning activities, including those that take place at the work site. As technologies and methods evolve and mature, the ability to identify and mitigate safety clashes is likely to remain beneficial and relevant. Given that attributes can be assigned and modeled in a binary fashion (i.e., present or absent), the algorithms can be robustly applied and simple user interfaces can be created.

**CHAPTER 4: APPLICATION OF MACHINE LEARNING TO  
CONSTRUCTION INJURY PREDICTION**

## **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 ran two state-of-the-art ML models, Random Forest (RF) and Stochastic Gradient Tree Boosting (SGTB), on 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 predict *injury type*, *energy type*, and *body part* with high skill ( $0.236 < RPSS < 0.436$ ), outperforming the parametric models found in the literature. The high skill reached suggests that construction safety features a non-random component and should be studied empirically like other natural phenomena, 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.

## **INTRODUCTION AND MOTIVATION**

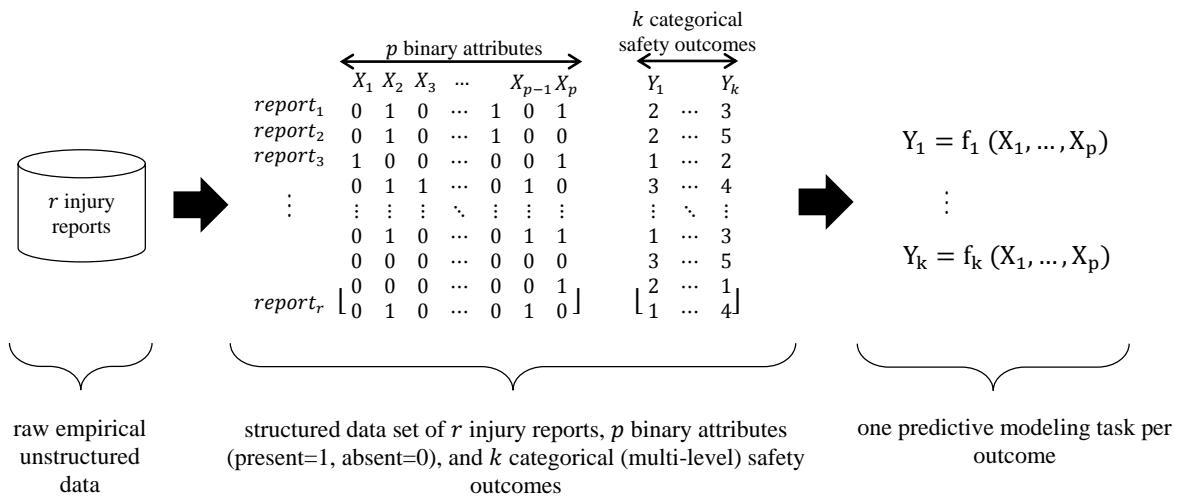
Construction is one of the largest industries in the United States, but is also one of the deadliest (Bureau of Labor Statistics 2013). Between 1992 and 2010, an average of 730 lives have been claimed every year (CPWR 2013). In addition to immeasurable human impacts, construction injuries also cost \$15 billion

annually. Despite the numerous efforts that have been motivated by this alarmingly poor performance, injury statistics have not improved significantly 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 structured safety critical 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 or project type. 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 current 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 injury 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 developing research area (e.g., Esmaeili et al. 2015a; Esmaeili et al. 2015b; Prades

Villanova 2014; Desvignes 2014; Esmaeli and Hallowell 2012, 2011b) is subject to limitations with respect to one or more of the aforementioned parameters.

Building on three recent studies (Prades Villanova 2014; Desvignes 2014; and Tixier et al. 2016a) that respectively addressed the limitations pertaining to the first three aforementioned criteria, 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 (Esmaeli et al. 2015b), both in terms of predictive skill and variety of outcomes predicted. These models provide actionable feedback that can be used to direct efforts towards targeted preventive actions and corrective measures. As will be described, these new models could be integrated with emerging technologies to yield safety forecasts at various stages of the project lifecycle.

## **BACKGROUND AND POINT OF DEPARTURE**

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

### **Why does safety outcome prediction matter?**

Many industries, including construction, struggle with decision-making under uncertainty. Making the wrong decisions can have terrible 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 indeed 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., Kahneman and Tversky 1982). 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 found safety-related decisions under uncertainty on empirical knowledge, which 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.

### **Limitations of previous work on attribute-based construction safety**

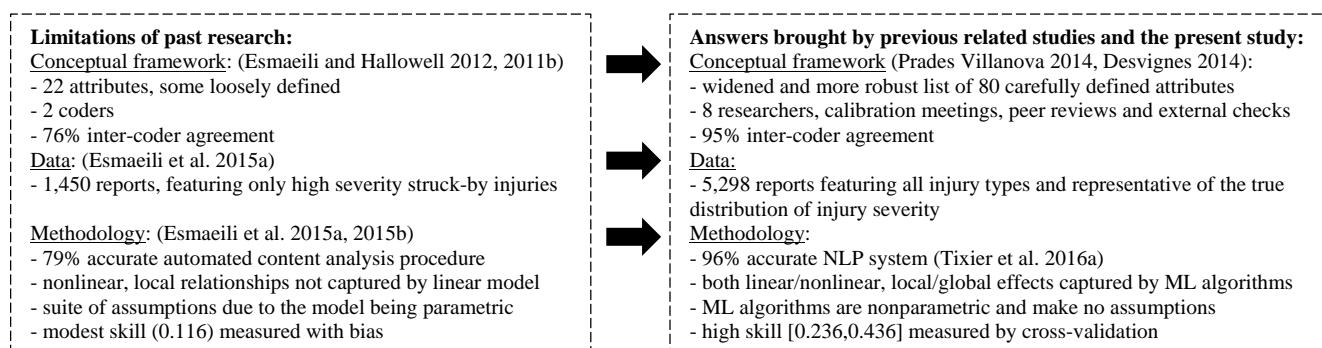
Although Esmaeili and Hallowell (2012, 2011b) made important strides by introducing and using the attribute-based framework for the first time, some serious limitations remained. In particular, some of the

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, as will be described in a following section.

To address the abovementioned limitations (which are summarized in Figure 2), we first used a broadened and more robust list of 80 attributes engineered and validated by a team of 8 researchers (Prades Villanova 2014, Desvignes 2014) and slightly modified by Tixier et al. (2016a). This list is provided in Table 2. Second, we used a rather large database of 5,298 injury reports featuring all types of injuries, and 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 96% accurate natural language processing (NLP) program developed by Tixier et al. (2016a), 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 (as objectively assessed by cross-validation). 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.



**Figure 2. Limitations of past research and solutions brought by the present study**

It should also be noted that predicting a binary severity outcome (fatality/not fatality) like in Esmaeili et al. (2015b) may not constitute the most useful safety forecast that can be issued. Indeed, the goal of construction safety management should be to prevent the occurrence of *all* injuries, without consideration for their supposed severity. In other words, if a safety issue has been identified, preventive measures should be taken regardless of whether there is a greater chance of observing a fatality, or say, a lost work time injury, according to some model. Consequently, predicting *injury type*, *body part* affected, and

*energy type* involved in addition to *injury severity*, like done in this study, may have greater practical value.

### **Previous use of machine learning (ML) in construction**

Construction research has considered ML for more than two decades. As soon as 1991, Moselhi et al. 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. Interestingly, while they listed optimization of activity and resource usage, and prediction of productivity, time and cost overrun as potential areas of neural network application, Moselhi et al. (1991) did not include construction safety outcome prediction in their list. However, Arciszewski et al. (1991) and Arciszewski and Usman (1993) acknowledged the potential of inductive learning for accident analysis and prediction, and conducted some preliminary experiments. Later, Skibniewski et al. (1997) conducted a study to demonstrate the feasibility of ML in constructability analysis. They applied the AQ15 algorithm on a collection of 31 training examples to automatically learn the mapping between the dependent categorical variable constructability (poor, good, excellent) and 7 independent variables, 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 real world construction management database to identify the causes of delays. They also recognized the needs to adopt new techniques to help humans extracting useful information from the explosive amount of digital data produced in the construction industry.

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 at

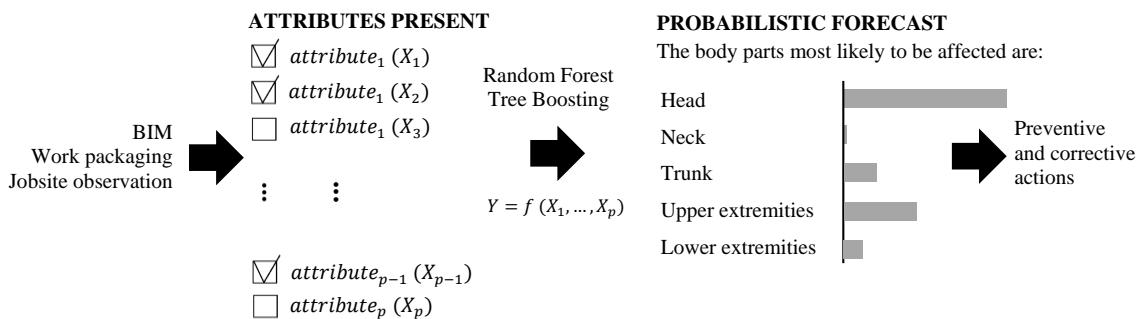
completion from ten input variables, such as change orders and number of rainy days, and to estimate the loss risk associated with a given construction project, in order to determine the optimal insurance deductible. The input variables included project duration, number of floors, construction season, and geological conditions. More related to our work, Rivas et al. (2011) applied several ML algorithms to predict a severity-related binary outcome: *accident* or *incident* occurrence. They used data obtained from 62 questionnaires (18 corresponded to accidents and 44 to incidents) by companies involved in construction and mining work in Spain. The questionnaires included a blend of 17 objective and subjective questions, such as time of the day, task duration, worker age, risk awareness, personal factors, or contractual status. The models reached good skill, with task duration and contractual status being the most predictive variables. In simple terms, Rivas et al. (2011) found that accidents occurs if the task lasts less than 4 hours and if the worker involved is a subcontractor. However, with such a limited amount of data, these findings are anecdotal. 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 far from being 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, except a few attempts in the early 1990s (i.e., the work of Arciszewski), this is only the third time that supervised learning algorithms are used to predict construction safety-related outcomes from empirical data (after Esmaeili et al. 2015b and Rivas et al. 2011).

## Goal of this study

The goal of the present research effort is to use Random Forest (RF) and Stochastic Gradient Tree Boosting (SGTB), two widely used and highly successful Machine Learning (ML) algorithms, on attribute and outcome data extracted from a large body of injury reports. 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 3 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 Villanova (2014) is useful, but predicting the most likely categories of various safety outcomes is a complementary and equally valuable strategy.



**Figure 3. Practical use of the predictive models built in this study**

## **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 Villanova (2014), Desvignes (2014), and Tixier et al. (2016a). These reports were automatically scanned for the attributes shown in Table 1 and the safety outcomes listed in Table 2 by a 96% accurate NLP system developed by Tixier et al. (2016a).

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) and the Occupational Safety and Health Administration (OSHA) (BLS 2010, Hallowell 2008).

It should be noted that Prades Villanova (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).

Also, Tixier et al. (2016a) 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 tool. At each round, lessons learned from examining the errors made by the system were used to improve skill. A harsh 95% threshold in accuracy was exceeded after 4 iterations (96%).

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

UPSTREAM*	n					
Cable tray	48	Rebar	155	Screw	37	
Cable	75	Scaffold	300	Slag	75	
Chipping	34	Soffit	12	Spark	9	
Concrete liquid	58	Spool	52	Slippery surface	142	
Concrete	165	Stairs	137	Small particle	401	
Conduit	56	Steel sections	759	Adverse low temperatures	123	
Confined workspace	129	Stripping	114	Unpowered tool	611	
Congested workspace	13	Tank	85	Unstable support/surface	8	
Crane	69	Unpowered transporter	53	Wind	109	
Door	85	Valve	79	Wrench	110	
Dunnage	29	Welding	200	Lifting/pulling/manual handling	553	
Electricity	3	Wire	131	Light vehicle	133	
Formwork	143	Working at height	268	Exiting/transitions	132	
Grinding	133	Working below elevated workspace/material	50	Sharp edge	47	
Grout	18	Drill	97	Splinter/sliver	41	
Guardrail/handrail	91	<b>TRANSITIONAL</b>		Repetitive motion	66	
Heat source	111	Bolt	186	Working overhead	14	
Heavy material/tool	79	Cleaning	119	<b>DOWNSTREAM</b>		
Heavy vehicle	143	Forklift	39	Improper body position	88	
Job trailer	24	Hammer	149	Improper procedure/inattention	57	
Lumber	252	Hand size pieces	172	Improper security of materials	87	
Machinery	189	Hazardous substance	156	Improper security of tools	28	
Manlift	66	Hose	95	No/improper PPE	23	
Stud	31	Insect	105	Object on the floor	174	
Object at height	86	Ladder	163	Poor housekeeping	2	
Piping	388	Mud	35	Poor visibility	12	
Pontoon	15	Nail	94	Uneven surface	59	
		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. (2016a)

**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			

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 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. 2016a).

900 reports out of the 5,298 available were not associated with any attribute, and were therefore removed. An inspection of these reports revealed 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 (2016a) NLP tool cannot extract the safety outcome injury severity, so for this prediction task, the 1,829 reports manually analyzed by Prades Villanova (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 level of the safety outcomes in the final data sets are reported 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	1,532	TOTAL	1,828
Motion	2,780	Struck by or against	2,401	Lower extremities	710		
Pressure	47	TOTAL	4,395	TOTAL	3,556		
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  $Y_1 = \text{energy source}$  (7 levels),  $Y_2 = \text{injury type}$  (5 levels),  $Y_3 = \text{body part}$  (5 levels), and  $Y_4 = \text{injury severity}$  (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.

## APPLICATION OF MACHINE LEARNING (ML)

After explaining why ML was preferred to more classical parametric modeling, the CART, Random Forest (RF) and Stochastic Gradient Tree Boosting (SGTB) algorithms are introduced, and we present and justify the combination of methodological choices made to address class imbalance and parameter optimization, and discuss the application of the procedures in practice

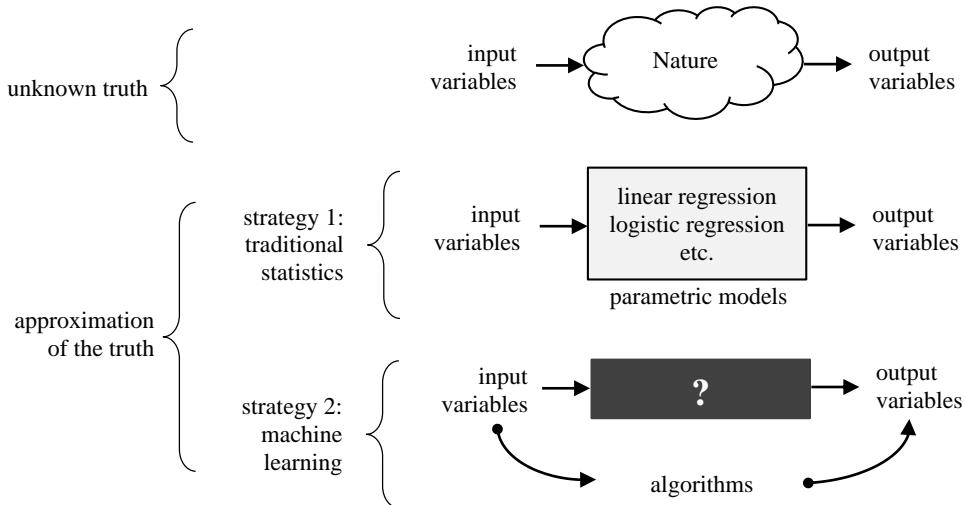
RF and SGTB were applied to the  $r = 4,398$  by  $p = 78$  structured data set of attributes and outcomes shown in Figure 1 ( $p = 78$  since *poor housekeeping* and *electricity* were removed as previously explained). The rationale for using two different algorithms stemmed from (1) the exploratory 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 and corroborating variable importance measures.

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). For instance, Esmaeili et al. (2015b) assumed that attributes and safety outcomes were related in a linear way, as they used logistic regression (a

Generalized Linear Model where the link is the logit function). This is obviously inadequate if the true underlying mechanisms relating attributes to outcomes happen to be nonlinear.

On the other hand, as shown in Figure 4, ML algorithms do not assume that the data have been generated by any parametric model prescribed upstream 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. For these reasons, ML was preferred to parametric modeling in this study.

As described in Figure 5, the ML task of interest here is that of *supervised* learning, that is, learning the associations between input and output variables from labeled training examples (Hastie et al. 2009, Chapter 2). On the other hand, principal component analysis, cluster analysis, and other similar data mining techniques are *unsupervised* learning methods because they group observations using input variables only, without taking into account any output variable.



**Figure 4: ML versus traditional statistics (adapted from Breiman 2001a).**

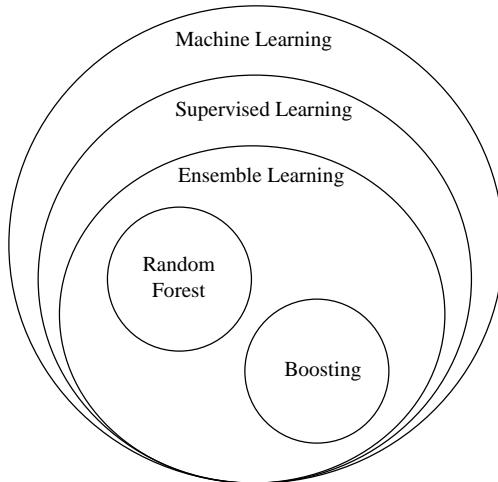
Here, the features, or input variables, were the fundamental construction attributes ( $X_1, \dots, X_{78}$ ) listed in Table 2, such as *welding*, *uneven surface*, or *adverse low temperatures*, and the targets, or output variables, were the four safety outcomes ( $Y_1, \dots, Y_4$ ), listed in Table 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, supervised ML algorithms could infer rules mapping combinations of attributes to safety outcomes, and use these rules later on to predict the most likely outcomes for brand new observations. There are two types of supervised learning problems: classification, when the response variable is categorical, and regression, when the response is continuous. In this study, all safety outcomes were categorical.

### **Ensemble learning**

Both Random Forest (RF) and Boosting are ensemble learning techniques. As illustrated in Figure 5, ensemble learning is a subset of supervised learning where committees of models are used in lieu of single models. Since their emergence in the 1990s, ensembles of models have proven to significantly outperform single models in terms of predictive accuracy, both for regression and classification problems. They are now standard and widespread (Brown 2010, Biau et al. 2008, Friedman and Popescu 2008, Opitz and Maclin 1999).

Ensembles differ in the base models they are built from, and in the way the predictions of the individual base models are aggregated to form an overall prediction. Boosting (Freund and Schapire 1995) and RF (Breiman 2001b) are two of the most popular and successful examples of ensembles (Robnik-Šikonja 2004). As the base model used in both RF and Stochastic Gradient Tree Boosting (SGTB) are built via the Classification and Regression Tree (CART) algorithm (Breiman et al. 1984), this algorithm must be

introduced first. Also, since RF is a simple improvement over the Bagging algorithm (Breiman 1996a), Bagging must be explained before RF can be tackled.



**Figure 5. The two methods used in this study, RF and Boosting, are both ensemble learning techniques. Besides that, they widely differ.**

### **Classification and regression trees (CART)**

CART is a greedy algorithm introduced by Breiman et al. (1984) that is used to learn (near-optimal) decision trees from the data. This process induces rules representing the underlying concepts in the data (Murthy 1995). As opposed to *global* models such as GLM, 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 relationship between input and output variables. Using binary splits, trees recursively partition the predictor space by identifying the regions that have the most homogeneous responses to predictors. Then, a constant is locally fit to each final region (or leaf): the most probable category for classification, and the average for regression (Elith et al. 2008).

More precisely, the steps of the CART algorithm are as follows:

- i. start with a root node that includes all observations,
- ii. for each predictor, compute a “goodness of split” value based on some criterion,
- iii. pick the best predictor (if multiple predictors are best, select the first one),
- iv. split the root node into two child nodes based on the values taken by the observations on the predictor selected at step iii,
- v. repeat steps ii to iv for each node created until maximum size is reached.

For classification, which is the aim of the present study, the criterion used at step ii is the decrease in partition impurity. A partition is considered 100% pure if it contains only observations belonging to the same class. More specifically, the best predictor selected at step iii is the one that maximizes the Gini gain, which is computed for a given predictor X and for a given node N as shown in equation 1 (Breiman et al. 1984). Note that for regression, the best predictor is the one maximizing the decrease in variance.

$$\text{GiniGain}(N, X) = \text{Gini}(N) - \frac{|N_1|}{|N|} \text{Gini}(N_1) - \frac{|N_2|}{|N|} \text{Gini}(N_2)$$

**Equation 1. Gini gain for a given predictor X, for a given node N, where  $N_1$  and  $N_2$  are the two child nodes induced by splitting N on X, and  $| . |$  denotes the number of elements in a node (its cardinality).**

The Gini diversity index  $Gini(N)$  is an impurity function that is used to characterize the heterogeneity of a node  $N$  as shown in equation 2:

$$Gini(N) = 1 - \sum_{k=1}^K p_k^2$$

**Equation 2. Gini diversity index for a node N, where K is the number of categories of the observations in the node, and  $p_k$  is the proportion of observations in the node falling in category k.**

The Gini index reaches its minimum (zero) for a completely homogeneous node featuring one single class, and reaches its maximum for a completely heterogeneous node where each observation belongs to a different category (Raileanu and Stoffel 2004). In other words, the Gini index for a node  $N$  is the probability that an observation randomly selected from node  $N$  is misclassified if it is randomly assigned to a class according to the class proportions in  $N$ .

Typically, a tree is grown until a certain predefined maximum number of nodes is reached, or when splitting does not lead to any significant decrease in partition impurity (classification case) or in node variance (regression case) for any node. In practice, trees often need to be grown very large to accurately represent their training data.

Some advantages of trees include their ability to capture complex nonlinear high-order interactions among predictors, to handle highly dimensional data sets with large numbers of observations, and their robustness to outliers (Hastie et al. 2009, Sutton 2005). As can be seen on the right of Figure 6, another advantage of trees is that they “explain” how they classify observations via a sequence of simple “IF-THEN” rules, in an intuitive way that is easy to grasp and represent. Finally, trees grown through CART are insensitive to the inclusion of irrelevant predictors, because such variables are naturally left aside during the tree-building process (see step iii of the CART algorithm described above). Therefore, trees do not need to be provided *a priori* with a best subset of covariates (like regression models for instance), as

they select the significant predictors on their own (Timofeev 2004). A side effect of this selection process becomes apparent when dealing with ensemble of trees though, and will be described later on, in the section dedicated to RF.

A major disadvantage of trees is that they need to be grown very large to reach high training set accuracy, at the risk of not discerning the noise in the training set from the true signal, trends, and underlying structure of the data. This phenomenon is known as overfitting (Louppe 2014). The extreme overfitting case corresponds to a tree that would have as many leaves as there are observations. Therefore, to prevent overfitting while conserving a decent training set accuracy, one needs to find the right trade-off between shallowness and depth. It is common practice to use stopping criteria, or to remove the least important splits after the tree has been fully grown, which is known as *pruning* (Hastie et al. 2009, p. 308). Despite these practices, trees remain highly dependent upon their training data, and the removal or addition of only a few training observations can modify the entire tree structure in a cascading fashion (Strobl et al. 2009, Elith et al. 2008, Timofeev 2004). From the perspective of the bias-variance framework, where  $\text{error} = \text{bias} + \text{variance}$ , trees have low bias, but high variance (Hastie et al. 2009, p.587). This makes for good accuracy on the training set, but for poor predictions.

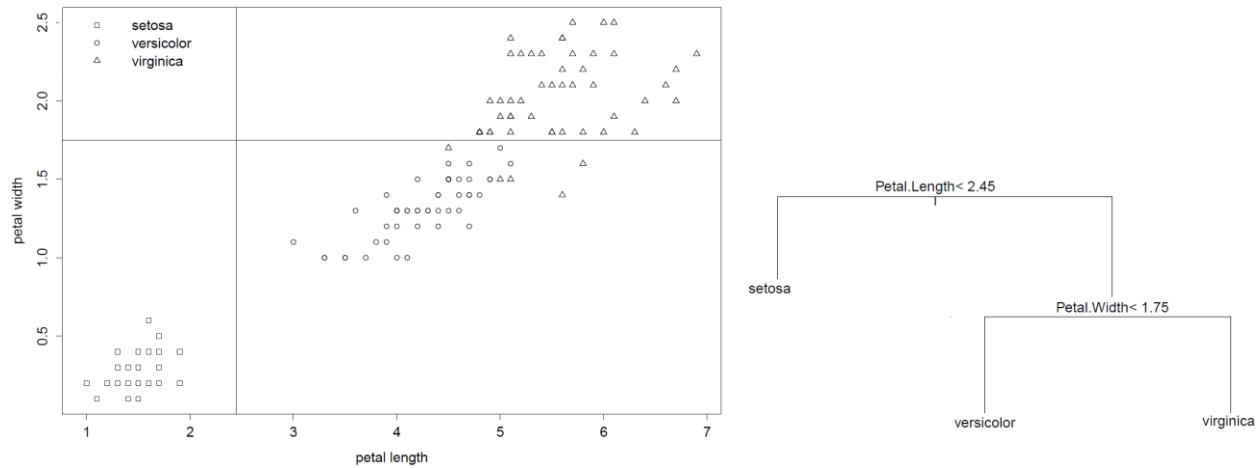
### **CART example**

Since understanding CART algorithm is crucial to the understanding of RF and SGTB, we provide an illustrative example in Figure 6. This example is based on the famous “iris” data set (UCI Machine Learning Repository, Linchman 2013), and uses CART for classification, as it is the task of interest in this study. 102 flowers have been measured for two input variables: *petal.length* and *petal.width*, and one output variable, *species*. The observations are plotted in this two-dimensional feature space on the left of Figure 6. The outcome *species* has three levels: Virginica (plotted as triangles, 45 observations),

Versicolor (circles, 35), and Setosa (squares, 22). The goal is to predict *species* based on the two features *petal.length* and *petal.width*.

By definition of CART, all features are considered at each split, and the split is made on the feature that maximizes the Gini gain as defined in equation 2. Making a split on this best predictor yields the purest (or less heterogeneous) partition of the observations into two groups. As one can see, both *petal length* and *petal width* are ideal candidates, as they allow to obtain a 100% pure partition.

Indeed, in each case, the Setosa flowers end up completely separated from their non-Setosa counterparts. In such cases, CART simply selects the first of the best predictors as found in the original data frame. Here, because the variables in the “iris” data set are ordered alphabetically, the first split is made on the predictor *petal.length*, as shown on the right of Figure 6. After this split is made, the left leaf is 100% pure, and requires no further attention. However, the right node is still heterogeneous as it comprises both Versicolor (circles) and Virginica (triangles) observations.



**Figure 6: CART example in the two-dimensional case**

Intuitively, one can see on the left of Figure 6 that making a horizontal split at `petal.width = 1.75` would result in 5 Virginica flowers being wrongly classified as Versicolor in the lower child node, but in all Versicolor flowers correctly classified in the upper child node. On the other hand, making a vertical split, say, at `petal.length = 4.85`, would yield a partition where 2 Virginica and 3 Versicolor flowers are missed, respectively in the left and right child nodes. Both splits are associated with the same overall error rate, but the split made on `petal.width` has the advantage of producing a pure node, which is highly valued by CART. Therefore, the predictor leading to the purest partition would intuitively be `petal.width`. By computing the gains in impurity decrease using equations 1 and 2, one could also prove that splitting on `petal.width` is best, since it maximizes the GiniGain function (0.383, versus 0.375 for a split made on `petal.length`).

It should be noted that by strictly following the CART algorithm, additional splits would need to be made until all nodes have reached maximum purity, or until a certain predefined minimum number of observations per node has been reached. However, the very large tree obtained would capture the peculiarities of the training data rather than the general structure that should be learned. This phenomenon, known as overfitting, would deteriorate prediction performance for future observations. By using pruning to limit tree complexity, one would find the tree shown on the right of Figure 6.

Finally, to predict the species of a new flower, the user simply needs to follow a top-down path through the tree by answering the binary questions asked at each split. The predicted class of the new observation is the average of the leaf it falls into (for regression), or its most frequent class (for classification). Here, we are in the classification case, the labels of the leaves of the tree shown on the right of Figure 6 therefore indicate their most frequent class.

## Bagging

Breiman (1996a) introduced the **Bootstrap AGGREGATING** (Bagging) method as a way to take advantage of the low bias of CART-grown trees while stabilizing their high variance (Louppe 2014). The Bagging procedure consists in training unpruned (i.e., very large) trees in parallel, each on a bootstrap sample of the data. A bootstrap sample is obtained by randomly selecting observations with replacement from the original training data set until a data set of the same size is obtained. Approximately one third of the observations are not expected to be present in each bootstrap sample, because the probability of not selecting a given observation with replacement from a sample of size  $n$  is  $\left(1 - \frac{1}{n}\right)^n$ , which tends to  $\exp(-1) \approx \frac{1}{3}$  when  $n$  tends to infinity. These observations compose what is called the “out-of-bag” (OOB) sample (Breiman 1996b). Since the bootstrap sample is of same size as the original data set, it follows that for a large number of observations, each bootstrap sample is expected to contain about two thirds of unique examples, the rest being duplicates (Brown 2010, Robnik-Šikonja 2004). This causes each tree in the ensemble to be an expert on some specific domains of the original training data set, while being incompetent elsewhere (Opitz and Maclin 1999). Bagging thus creates an ensemble of local experts.

Consequently, when tasked with predicting the category in which a new observation should fall, there will be a significant amount of beneficial disagreement among trees. By aggregating the predictions from all the trees in the ensemble (average in the case of regression, majority vote in the case of classification), one gets a model that exhibits significantly less variance than a single unpruned tree (indeed, the original variance is divided by the number of trees in the optimal case), and therefore generalizes much better, while still having (almost) the same low bias. This approach was described by Breiman (1998) as “perturb and combine”. The only tuning parameter of Bagging is  $ntree$ , the total number of trees in the ensemble.

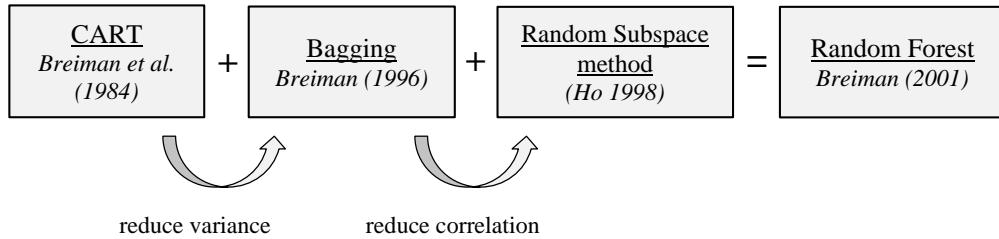
### ***OOB error estimate***

A direct and important benefit of having an OOB sample is the ability to monitor the predictive accuracy of the ensemble as it is being built (Breiman 2001b). To be more precise, every time a new tree is added to the ensemble, any observation in the training data set can be passed to all the trees in the ensemble that were trained on bootstrap samples devoid of this observation. As already explained, approximately one third of the total number of trees meet this condition. Further, a prediction can be obtained by aggregating the individual predictions of these trees (i.e., most frequently predicted class in the case of classification). This process is repeated for all observations in the training set, and the average misclassification rate, also called the OOB error estimate, can be computed for each category. This mechanism, very similar to N-fold cross-validation (Hastie et al. 2009, p. 593), was shown to be an unbiased estimator of predictive error, and to be as accurate as using a separate test set of the same size as the training set (Wolpert and Macready 1999, Breiman 1996b).

Despite being a significant improvement over CART, Bagged ensembles are less interpretable. Also, what is problematic is that by definition of CART, only those variables yielding the greatest decrease in node impurity are selected at each split. Consequently, except for the less important bottom splits (which capture more noise than signal), all the trees in the Bagged ensemble have quite similar structure, and therefore tend to generate correlated forecasts. This phenomenon reduces disagreement among trees, which diminishes the benefit of majority voting. Indeed, majority voting is only effective to the extent that there is disagreement (Opitz and Maclin 1999). Quantitatively, the correlation among trees prevents the maximum reduction in variance (i.e., variance divided by the total number of trees) to be achieved. This motivates RF as explained in what follows.

## Random Forest (RF)

Inspired by Ho's (1998) random subspace method as illustrated in Figure 7, Breiman (2001b) introduced Random Forest (RF) to address the correlation issue of Bagging previously described.

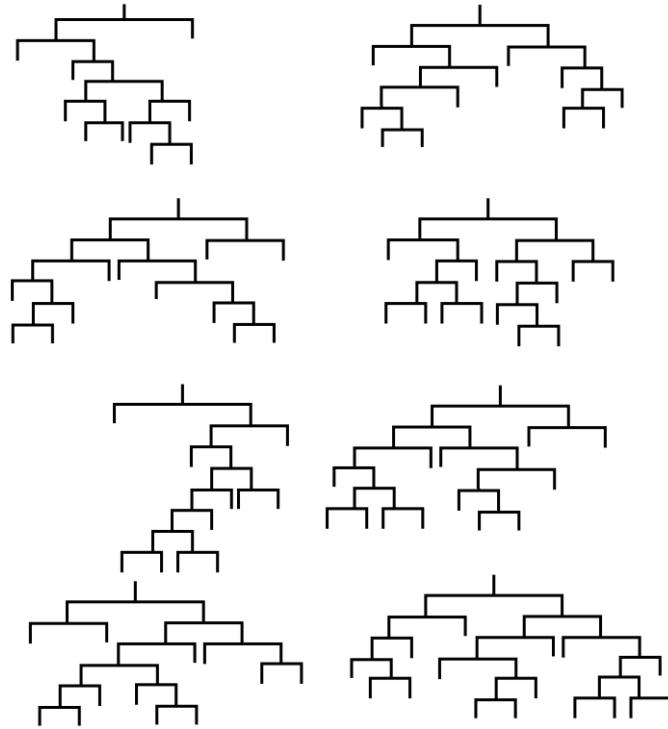


**Figure 7: From CART to Random Forest**

RF grows trees via a simple modification of the CART algorithm: instead of trying all predictors at each split, only a random subset of constant, predetermined size is considered. Note that trees are still grown on bootstrap samples of the training set, just like in Bagging. In practice, this additional injection of randomization gives all predictors a chance to play a role in determining the upper structure of trees, which introduces a lot of variety in the ensemble, de-correlates the trees, and allows disagreement and majority voting to be really leveraged. This results in greater variance reduction, smaller error rates and more accurate predictions as compared to Bagging. A schematic diagram of a small RF model is provided in Figure 8.

In addition to sharing the *ntree* parameter with Bagging, the RF model has another tuning parameter, called *mtry*, which determines the number of predictors that are randomly considered as candidates at each split. Setting low values of *mtry* increases the importance of randomization, and vice versa. If there are  $p$  predictors, setting  $mtry = p$  is equivalent to Bagging. Genuer (2008) notes that the best values of *mtry* and *ntree* are really dependent on the data and prediction task at hand, and that these parameters should systematically be tuned. The tuning protocol that we used will be discussed in detail in a

subsequent section. The “randomForest” package (Liaw and Wiener 2002) of the R programming language (R Core Team 2015) was used to build all the RF models.



**Figure 8. Schematic representation of a small Random Forest ( $ntree = 8$ ).**

Tree are independent, grown in parallel, and very deep due to the absence of pruning. Each tree fits its training data (unique bootstrap sample of the original training set) very well and has very low bias but high variance. The RF algorithm reduces variance by creating diversity in the upper structures of trees and by aggregating their individual, uncorrelated predictions via majority voting.

We used RF because it stands among the most accurate general-purpose classifiers to date (Biau 2012), and has shown great effectiveness in a variety of other fields. To cite only a few examples, the RF algorithm has been used to predict patient risk for various diseases (Lebedev et al. 2014, Khalilia et al. 2011), identify central genes (Díaz-Uriarte and Alvarez De Andres 2006), 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).

Interestingly, while the overarching principle and intuition are relatively straightforward, the exact reasons why RF produces such good results in practice are not fully understood yet from a theoretical standpoint, and the underlying statistical mechanisms remain largely unknown and are still under active investigation (Biau et al. 2008).

### ***Variable importance measure***

In many situations, high predictive accuracy is necessary but not sufficient: it is often very helpful to gain additional insight by understanding which variables bring predictive power. To fulfill this need, Breiman (2001b) showed that the out-of-bag (OOB) observations can also be used to compute a relative importance score for each predictor. More precisely, for a given predictor, and for a given tree in the forest, the procedure consists in randomly permuting the values of the predictor in the set of observations that have not taken part in the training of the tree (i.e., the OOB sample), and comparing the prediction error of the tree on the permuted OOB sample with the prediction error of the tree on the untouched OOB sample. This process is repeated for all the trees in the forest, and the predictor is given an importance score proportional to the overall increase in error that its permutation induced. The most important variables are the ones leading to the greatest losses in predictive accuracy when “noised-up” (Breiman 2001b).

### **Boosting**

Like Random Forest, the Boosting algorithm is an ensemble approach that combines many base models and let them vote to generate forecasts (Freund et al. 1999). This apparent similarity is misleading, since

RF and Boosting tackle the task of error reduction (where  $error = bias + variance$ ) in radically opposite ways (Hastie et al. 2009, p.337). Indeed, while RF seeks to reduce error by decreasing the variance of complex “low bias-high variance” base models (i.e., large CART-grown decision trees), Boosting achieves the same goal by reducing the bias of weak “high bias-low variance” base models. One should note that by averaging the output of many models, Boosting also reduces variance (to a lesser extent) in addition to bias, whereas in RF, bias cannot be reduced, and the only hope in improving accuracy lies in variance reduction (Hastie et al. 2009, p. 588). This could explain why Boosting is considered to have a slight edge over RF in many practical situations. However, as will be explained, Boosting has more parameters than RF and therefore requires costlier optimization, while RF is more “off-the-shelf”. In addition, the parallel nature of RF makes it ideal to harness parallel computing, whereas Boosting is sequential and thus tougher to parallelize.

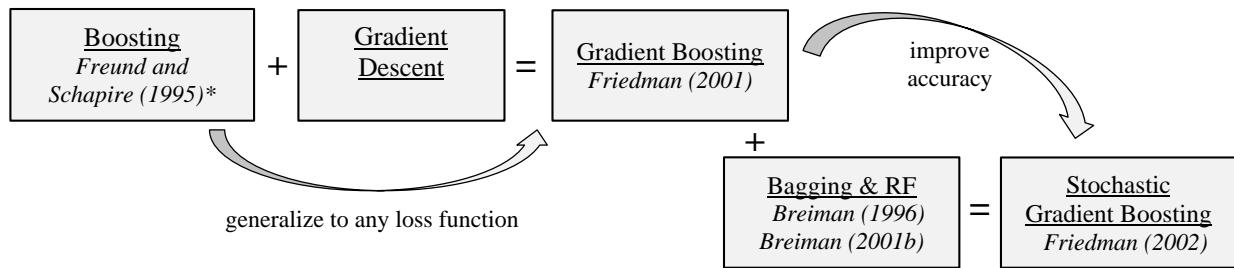
Like RF, Boosting is often used with decision trees, as it has proven extremely effective when used with this base model (Hastie et al. 2009, p. 340). In that case, Boosting is sometimes called Tree Boosting or additive trees, but since there is no ambiguity here, it will simply be referred to as Boosting in what follows, for brevity. Even though both algorithms use trees as their base models, it should be stressed once more that RF aggregates the output of many deep (unpruned) trees, whereas Boosting builds a sequence of small trees, sometimes as shallow as decision stumps (trees with two leaves).

Because it turns 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). Indeed, why struggling in devising costly and complex strong classifiers, when similar or even better skill can be reached simply by combining many cheap weak classifiers? The key lies in adding each weak classifier in sequence, such that each successive one focuses on capturing the regions of the training set that were missed by the preceding classifier. In other words, while RF relies on randomization and the law of large numbers to decrease variance through averaging

independent low-bias trees, Boosting inherently de-correlates low-variance trees by having them focus on different areas of the training set (the ones that were missed by the previous trees in the sequence), in order to decrease the bias of the series of trees as a whole. However, one should note that the fact that predictions of trees are de-correlated does not mean that they are independent. On the opposite, trees in Boosting all recursively depend upon one another.

### ***AdaBoost***

As shown in Figure 9, the first applicable and successful Boosting algorithm, AdaBoost, is due to Freund and Schapire (1996). In this algorithm orientated towards binary classification, all observations are initially assigned the same weight. A small tree is fitted to these data, and the observations that are misclassified by this tree see their weights increase, whereas the observations that are correctly categorized see their weights decrease. Every time a new tree is added to the model, the weights are updated. As a result, the observations that are repeatedly being missed throughout the process become the center of attention until they are correctly classified (Freund et al. 1999).



**Figure 9: from Boosting to Stochastic Gradient Boosting**

\* the origins of Boosting can be tracked back to anterior works. However, Freund and Schapire (1995) were the first to propose a successful implementation of a Boosting algorithm (AdaBoost).

The final model obtained is a nested sequence of small trees that all recursively depend upon each other, as shown in Figure 10. This is the opposite of RF, where independent trees are grown very large. Boosting has more parameters than RF (5 versus 2). The first is the number *n.tree* of trees in the sequence. A high number of trees are needed to achieve good learning, but unlike with RF, having too many trees can lead to overfitting on noisy data sets (Opitz and Maclin 1999), so close monitoring of the *n.tree* parameter is indispensable.

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), as shown in Figure 10, allow first-order (two-variable) interactions to be captured, and so forth (Hastie et al. 2009, p. 362).

### ***Gradient Boosting***

Five years after the introduction of AdaBoost, it was discovered that this technique was actually equivalent to estimating the parameters of a generalized additive model using a forward stagewise optimization strategy based on the minimization of an exponential loss function (Friedman et al. 2000). This discovery opened the way to improvement. Friedman (2001) generalized Boosting to be used with any differentiable loss function, which allowed the exponential loss function (not very robust under noisy conditions such as mislabeling of training examples) to be replaced with more robust functions, like the binomial deviance (Hastie et al. 2009).

This method was named Gradient Boosting, as it minimizes the loss function via a numerical local optimizer, gradient descent. However, while AdaBoost fits each tree of the sequence on a reweighted version of the original data (due to the special form of the exponential loss), Gradient Boosting fits each new tree directly on the gradient of the loss function of the current model, made of all the trees so far in the sequence (Friedman 2001).

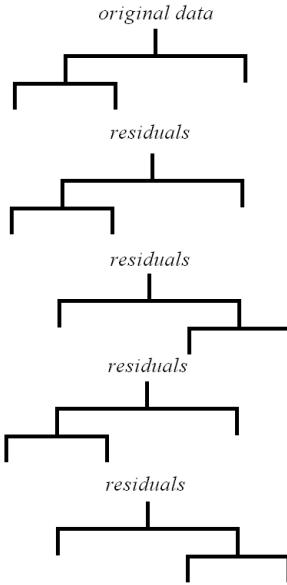
For least squares Boosting, that is, when the response variable is continuous and the function to minimize is the squared error, the intuition is really easy to get. At any given step, as illustrated in Figure 10, a new tree is simply fitted to the residuals of the current model. Then, this new tree is added to the model, the residuals are updated, and the algorithm continues to iterate.

An important parameter introduced here is the *learning.rate*, also known as *shrinkage*, which is a factor between 0 and 1 that shrinks the contribution of each new tree added in the series. It acts as a second regularization parameter, the first being *n.tree*, the total number of trees 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).

This makes sense, because with small increments, the algorithm can approach the minimal value of the loss function more closely. However, in return, the optimization process takes more time. A fourth parameter is *n.min*, the minimum number of observations allowed per node. By impacting the size of the trees, this parameter also acts as a regularization parameter. Larger values of *n.min* generate smaller trees, which are less influenced by noise.

### ***Stochastic Gradient Tree Boosting (SGTB)***

Finally, motivated by the work of Breiman on Bagging and Random Forest (e.g., Breiman 1996a, Breiman 2001b, respectively), Friedman (2002) slightly modified the Gradient Boosting algorithm so that at each round, 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. This method was named Stochastic Gradient Boosting, to emphasize the instillation of randomness into the procedure.



**Figure 10. Schematic representation of Least Squares Gradient Boosting with  $n.\text{tree} = 5$ .**

The trees are very small: they have only three final nodes, and all depend on each other. Each tree is a weak learner, in that it is only slightly better than random guessing. Thus, each tree has low variance but high bias. Bias is reduced by sequentially adding trees where each subsequent tree captures the observations that were missed by the preceding one. To a lesser extent, variance is also reduced by averaging many trees.

The necessity to adjust the proportion of training examples randomly drawn at each round adds one more tuning parameter, called *bag.fraction*. Randomization was found to lead to significant improvements in accuracy, although the exact cause of this improvement could not clearly be isolated (Friedman 2002).

In this study, SGTB models were created with the “gbm” R package (Ridgeway et al. 2015). The procedure used for parameter tuning will be detailed in a subsequent section.

#### **Variable importance measure**

Just like with RF, the Boosting algorithm allows the calculation of importance scores for the predictor variables (Elith et al. 2008). In the classification case, the procedure is as follows. For a given tree in the sequence, and for a given non-terminal node of this tree, the reduction in node purity as computed by

*GiniGain* (see equation 1), weighted by the number of observations in the node, is attributed to the predictor the split was made on. This process is repeated for every non-terminal node of the tree, and the variable importance scores are averaged over all trees. Finally, all scores are scaled by the number of observations in the training sample.

The next section discusses an important issue that arises when applying ensemble learning techniques on imbalanced data sets.

## Class imbalance issue

### Description of the problem

Our data set featured some significantly underrepresented categories, which is also commonly observed 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). What is really problematic in imbalanced learning is not so much the relative between-class imbalance itself, but rather that this phenomenon often goes hand in hand with *absolute rarity* of the minority class training examples (He and Garcia 2009, Weiss 2004). In this research, the problem arose in all prediction tasks. 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 significantly 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 is a critical issue since in most practical

applications, the rare classes are precisely the classes of interest: person at risk for contracting a rare disease, defaulting borrower, fraud... In this study, accurately predicting the rare categories, such as for example *chemical* for energy type, or *caught in or compressed* for injury code, was at least as important as predicting the majority classes like *motion* or *struck by or against*.

### ***Solutions found in the literature***

Addressing the class imbalance issue has motivated extensive research. Nevertheless, all the solutions proposed come down to either modifying (1) the learning algorithms, (2) the data themselves, or (3) both.

An example of a method belonging to the first category is Weighted RF (Chen et al. 2004), which is based on a modified version of CART where minority class examples are associated with higher misclassification cost, and where predictions are issued by weighted majority vote. Unfortunately, the version 4.6-10 of the “randomForest” R package (Liaw and Wiener 2002) that we used did not feature, at the time of writing, a functional version of Weighted RF algorithm.

On the other hand, the techniques based on modifying the data (resampling) generally involve oversampling the minority category or undersampling the majority category (del Rio et al. 2014). In its most basic form, oversampling reduces between-class imbalance by simply duplicating observations selected at random from the minority class, while undersampling discards observations at random from the majority category (Chawla 2002). For instance, Balanced RF (Chen et al. 2004) grows each tree on a bootstrap sample featuring the same number of observations from each class (where this number is the number of observations in the minority category). This procedure is called stratified undersampling. Chen et al. (2004) found that Weighted and Balanced RF were comparable in terms of performance, and that no clear winner emerged. Indeed, if both oversampling and undersampling improve model accuracy, both methods have disadvantages (He and Garcia 2009, Weiss, 2004, Japkowicz 2000). The most notable ones

are that oversampling can lead to overfitting (Drummond and Holte 2003, Chawla et al 2002), while undersampling throws information away (Chen et al. 2004).

To overcome these limitations, more sophisticated resampling techniques were proposed. A well-known example is the SMOTE algorithm (Chawla et al. 2002), which combines random undersampling with nearest-neighbor-based generation of synthetic minority training examples. SMOTE proved better than the basic resampling methods in certain situations, however, it is limited to binary classification tasks and could therefore not be applied to the multi-class classification tasks faced in this study.

### ***Approach used***

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, while preserving all the information from the majority categories. This strategy was implemented in R using the *sampsize* argument of the “random.Forest” 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 class priors (i.e., the probabilities of randomly drawing examples from each class).

Oversampling was preferred because it proved superior to undersampling with our data. This probably can be explained by the fact that because some minority categories comprised very few observations, undersampling led to too much information loss for the other classes. For instance, there were only 61 training examples available for the body part *neck*, whereas 899, 354, 1532, and 710 observations were respectively available for the categories *head*, *trunk*, *upper extremities*, and *lower extremities*. Therefore,

randomly drawing only 61 cases from each category (balanced undersampling) would have made use of all the cases from *neck*, but would have left out a considerable number of training observations. For *body part*, 96% of the observations from *upper extremities* would have been thrown away).

To conclude this section, 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 too 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.

### 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, bias and variance, or overfitting and underfitting (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

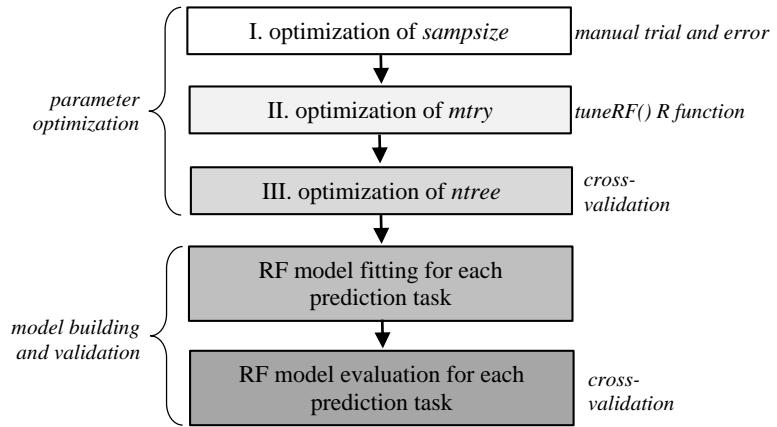
resources available, and the nature of the ML algorithm (Claesen and De Moor 2015). In what follows, we describe the approach we adopted to tackle parameter optimization for RF and SGTB.

### ***Parameter optimization for Random Forest (RF)***

As was previously mentioned, the RF algorithm has two tuning parameters: *mtry*, the number of variables randomly tried at each split, and *ntree*, the number of trees in the forest. In addition, as was also already explained, stratified oversampling was used to address class imbalance, introducing a third parameter, *sampsize*, controlling the class sampling proportions.

Jointly optimizing *sampsize*, *mtry*, and *ntree* using, say, an exhaustive grid search, would have been way too time consuming and computationally intensive. For instance, simply trying 8 values for *mtry* and *ntree*, 3 different sizes for each sampling strata (which is unrealistically low), and comparing between models based solely on 10 runs of cross-validation, would have required  $(3^5 \times 8^2 \times 10 \times 20)/3600 = 864$  hours of computing time, assuming five classes and 20 seconds to grow each RF model (which again, is rather optimistic for data sets featuring thousands of observations, large values of *mtry*, and hundreds of trees). Even using a 36 core machine, this still represents 24 hours of computing time. As noted by Claesen and De Moor (2015), optimization sometimes requires days or even weeks.

Because such resources were unavailable, *sampsize*, *mtry*, and *ntree* were optimized in sequence, as illustrated in Figure 11. The first step of the optimization procedure consisted in determining the best stratified bootstrap proportions (*sampsize*), and is detailed next.



**Figure 11. Overview of the parameter tuning and model evaluation procedure for RF**

#### *Step 1: optimization of the sampsize parameter*

Figure 12 shows the procedure followed to determine the best 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). Rounded to the nearest integer, the initial weights for this safety outcome were therefore  $1532/61 = 25$  for *neck*,  $1532/899 = 2$  for *head*,  $1532/354 = 4$  for *trunk*,  $1532/1532 = 1$  for *upper extremities*, and  $1532/710 = 2$  for *lower extremities*.

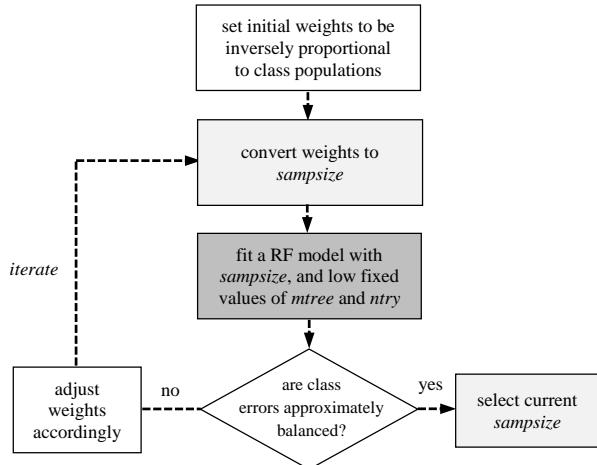
Randomly drawing with replacement from each class according to these weights generated bootstrap samples of the original training set where each class was approximately equally represented. Continuing with the *body part* example, the numbers of observations sampled from each category were:  $25 * 61 = 1,525$  for *head*,  $899 * 2 = 1,798$  for *neck*,  $354 * 4 = 1,416$  for *trunk*,  $1532 * 1 = 1,532$  for *upper extremities*, and  $710 * 2 = 1,420$  for *lower extremities*, making for initial bootstrap samples of 7,691 observations, where classes were represented roughly with equal proportions (one fifth each).

Finally, based on the “out-of-bag” (OOB, Breiman 1996b) 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 12, 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 was proven to be unbiased and at least as accurate than cross-validation (Wolpert and Macready 1999, Breiman 1996b).

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.



**Figure 12. Optimization of the *sampsize* parameter for RF**

**Table 4. Optimal weights and values of the *sampsize* parameter for each prediction task (RF)**

body part	head	neck	trunk	upper extremities	lower extremities	total
size	899	61	354	1532	710	3,556
weights	1.5	7.26	3.2	1.07	1.45	
sampszie	1348	443	1133	1633	1030	5,587
energy type	biological	chemical	gravity	mechanical	motion	pressure
size	108	197	1030	74	2780	47
weights	6.5	3.5	3.13	9.5	1.17	14.74
sampszie	702	690	3219	703	3239	4.5
injury type	caught	exposure	fall	overexertion	struck	total
size	334	496	570	594	2401	4,395
weights	5.25	1	2.25	5.5	1.5	
sampszie	1753	496	1282	3267	3602	10,400
severity	pain / first Aid		medical case		lost work time	total
size	1521		206		101	1,828
weights	1		4.66		6.66	
sampszie	1521		960		672	3,153

### **Step 2: optimization of the *mtry* parameter**

The function “tuneRF” from the “randomForest” R package (Liaw and Wiener 2002) was used to determine the best value of the *mtry* parameter, with arguments *stepFactor* = 1.2, *improve* = 0.01, and *ntreeTry* = 100. This optimization process, quite similar in spirit to gradient ascent (or gradient descent), can be described as follows:

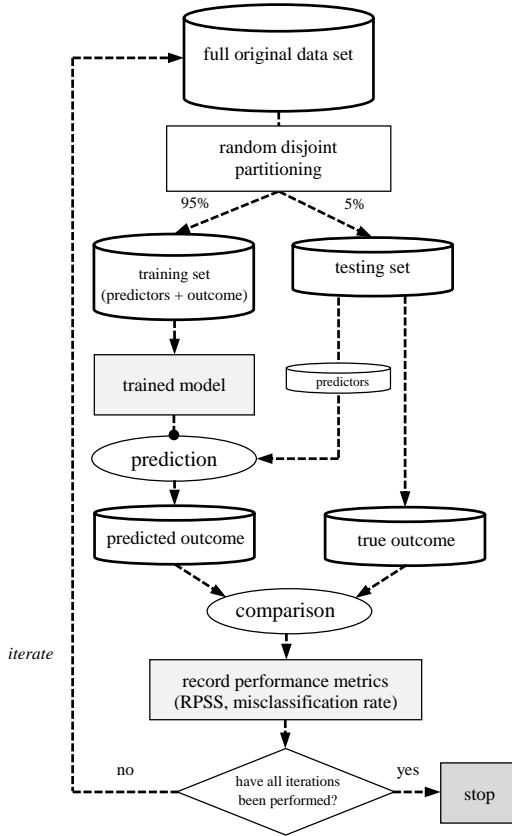
1. take the initial value of *mtry* to be the largest integer not greater than the default value  $\sqrt{p}$  recommended by Breiman (2001b) for classification,
2. fit a RF model with this initial value of *mtry*, and record the out-of-bag (OOB) error estimate,
3. determine the best search direction by looking to the left (largest integer not greater than  $\sqrt{p}/\text{stepFactor}$ ) and to the right (largest integer not greater than  $\sqrt{p} \times \text{stepFactor}$ ) of the initial value of *mtry*, fitting a RF model for each direction (each candidate value of *mtry*), and selecting the direction (the value of *mtry*) that maximizes the gain in OOB error reduction,

4. a. do not start the search if none of the directions leads to a decrease in OOB error greater than the *improve* parameter. In that case, select the initial value  $\sqrt{p}$  as the best value of *mtry*,
  - b. otherwise, conduct the search in the best direction, by iteratively fitting one RF model for each successive value of *mtry*, and recording the OOB error,
5. stop when iterating (i.e., dividing by *stepFactor*, for searches to the left, or multiplying by *stepFactor*, for searches to the right) does not yield a reduction in OOB error greater than the *improve* parameter, and return the final value of *mtry* as the best value.

The best direction was always the right. Since there were  $p = 78$  attributes, the values successively tried were the largest integer not greater than  $\sqrt{78}$ , that is, 8; the largest integer not greater than  $8 \times 1.2$  (9); and so forth, which gave 10, 12, 14, 16, 19, etc. The best values of *mtry* returned are shown in Table 5.

### *Step 3: optimization of the ntree parameter*

Eight different values of *ntree* (101 to 801, by 100) were compared based on 36 runs of "leave-5%-out" cross-validation. The proportion of training examples left out was set to 5% (rather than 10% or 20%) in order to avoid discarding too many training observations from the minority classes at each run. Cross-validation (Hastie et al. 2009, section 7.10) is a general and standard procedure used to optimize the parameters and objectively estimate the predictive skill of any model (Kohavi 1995). It works as described in Figure 13.



**Figure 13. “Leave-5%-out” cross-validation procedure**

More precisely, 5% of the observations were randomly put aside (without replacement) from the full data set at each round. This set of observations constituted the testing set. The model was trained on the remaining observations, called the training set. It should be emphasized that the training and the testing were mutually exclusive (as is always the case with cross-validation). The model learned the mapping between the input variables (i.e., the predictors) and the target variable (i.e., the safety outcome) from the training set. Then, the model was provided with the predictor portion of the testing set and asked to predict the target variable. Predictive skill was then evaluated by comparing the probabilistic forecasts that had been generated by the model to the known true values of the target variable. As will be discussed in a following section, predictive skill was measured in terms of the Rank Probability Skill Score (RPSS, Wilks 1995). Table 5 summarizes the best combinations of parameter values for each prediction task.

**Table 5. Optimal parameter values for each prediction task (RF)**

	<i>mtry</i>	<i>n.tree</i>
energy type	44	201
injury type	37	701
body part	31	601
injury severity	26	701

### **Parameter optimization for Stochastic Gradient Tree Boosting (SGTB)**

As previously explained, SGTB required the selection of an appropriate loss function, and the tuning of five parameters: the (1) number of trees in the sequence *n.tree*, the (2) maximal order of interaction that can be captured *interaction.depth*, the (3) minimum number of observations in each leaf *n.min*, the (4) *learning.rate*, and (5) the proportion of observations that are drawn at random from the original data set to grow each tree of the sequence, called *bag.fraction*. The loss function appropriate for the multiclass classification problems of this study was the multinomial deviance (Ridgeway 2012).

#### **Step I**

As shown in Figure 14, all parameters except *n.tree* and the oversampling proportions were first set to values recommended by the literature. In theory, the value of *interaction.depth* should be chosen to reflect the true order of interaction prevailing in the underlying process studied. However, most of the time, it is unknown (Hastie et al. 2009 p. 363, Elith et al. 2008), and this research was no exception. Because in practice, low order interactions tend to dominate, capturing them is generally sufficient to explain most of the interplay between input and output variables (Hastie et al. 2009 p. 363, Friedman 2001). Also, it was empirically shown that values between 4 and 8 give best results, and that all the values in that range can be considered equivalent (Hastie et al. 2009, p. 363). Therefore, *interaction.depth* was set to a value of 5.

The *bag.fraction* parameter was set to 0.5 for all prediction tasks since it was found in practice that the best values for this parameter were constantly around 0.5 (Ridgeway 2007, Elith et al. 2008, Friedman

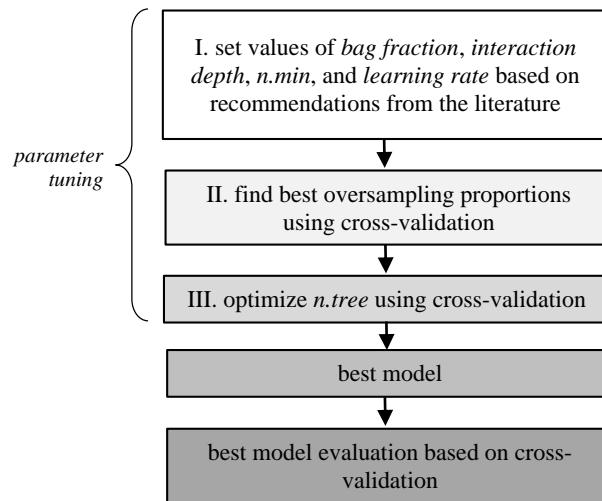
2002). Experiments with neighboring values did not yield any improvement in accuracy, corroborating our choice. Furthermore, following Ridgeway (2007), the *learning.rate* parameter was set to 0.005 as this value was reasonably low while still being computationally feasible. Indeed, slowing down the *learning.rate* significantly increases storage requirements and computation time. For the safety outcome *injury severity*, 0.005 was too slow, so the *learning.rate* was set to 0.01. Finally, a standard value of 5 was used for *n.min*, the minimum number of observations allowed per leaf.

### **Step II**

At step II, oversampling was used to address the class imbalance issue previously explained. Starting with all classes equal in terms of number of observations, oversampling proportions were adjusted (i.e., cases were duplicated) until the misclassification rate was approximately equally shared among all classes. Combinations were compared on the basis of 16 runs of leave 5% cross-validation, with an affordable value of *n.tree* (1200) that ensured approximate convergence without risking to overfit. The best sampling proportions found are summarized in Table 6.

### **Step III**

Finally, at step III, the *n.tree* parameter was optimized. We followed best practice which consists in finding the optimal number of trees by cross-validation after the value of the *learning.rate* has been set (Hastie et al. 2009, p. 365; Ridgeway 2007; Friedman 2001). This step was implemented by using the R “gbm” function (Ridgeway et al. 2015) which offers internal cross-validation (we used 8-fold cross-validation here). A sufficiently large initial value of *n.tree* was prescribed in order to let the “gbm” function find the inflexion point when the models began to overfit the data. This stopping value corresponded to the optimal tradeoff between goodness of fit and generalization ability. The optimal parameter values found for each prediction tasks are summarized in Table 7.



**Figure 14. Parameter optimization and model evaluation procedure used for SGTB**

**Table 6. Optimal resampling proportions and final numbers of cases in the resampled data sets for each prediction task (SGTB)**

body part	head	neck	trunk	upper extremities	lower extremities	total
original proportions	899	61	354	1532	710	3,556
weights	1.33	8	3.33	1	1.33	
resampled proportions	1200	488	1180	1532	947	5,346
energy type	biological	chemical	gravity	mechanical	motion	total
original proportions	108	197	1030	74	2780	47
weights	1	3	6	15	2	20
resampled proportions	108	591	6180	1110	5560	940
injury type	caught	exposure	fall	overexertion	struck	total
original proportions	334	496	570	594	2401	4,395
weights	11	1	3	6	2.33	
resampled proportions	3674	496	1710	3564	6403	15,847
injury severity	pain / first aid			medical case	lost work time	total
original proportions	1521			206	101	1,828
weights	1			6	8	
resampled proportions	1521			1236	808	3,565

**Table 7. Optimal parameter values for each prediction task (Boosting)**

	interaction depth	bag fraction	learning rate	n. min	n. tree
energy type	5	0.5	0.005	5	1200
injury code	5	0.5	0.005	5	1550
body part	5	0.5	0.005	5	900
injury severity	5	0.5	0.01	5	4000

## Measuring predictive skill with RPSS

We used the Rank Probability Skill Score (RPSS, Wilks 1995) to evaluate the predictive skill of the models. The RPSS is a metric widely used in climatology where probabilistic forecasts are common. Such forecasts, as illustrated in Figure 2, assigns a probability of occurrence to each level of the output variable instead of providing a single “best guess” prediction. Because it strongly penalizes confident forecasts of the wrong categories, the RPSS can be considered to be a stringent test of model performance (Goddard et al. 2003). In this study, using this metric was even more harsh because it assumes the categories to be ordered (e.g., low, medium, high), and penalizes forecasts more severely when their probabilities are further from the actual outcome (Franz and Sorooshian 2002). In other words, if the true observation is “high”, the RPSS penalizes more a model predicting “low” than a model predicting “medium”. This makes sense, but in this study, the classes (except for injury severity) were not ordered. For instance, considering the safety outcome *energy type*, a model assigning the greatest probability of occurrence to the category “chemical” should not be penalized more than a model predicting “biological” if the true outcome is “gravity”. Both models are equally wrong. The “rps” function from the “verification” R package (NCAR - Research Applications Laboratory 2015) was used to compute the RPSS.

As one can see from equation 3, the RPSS compares the predictions issued by a model to that of some reference, by taking the ratio of the average Rank Probability Score (RPS) of the forecasts generated by the model and the average RPS of the reference. The reference against which the model is compared can be random guessing (equal probabilities for each class), or be chosen to match the class probabilities empirically observed in the data or in nature (Franz and Sorooshian 2002). In this research, due to the severe between-class imbalance issue previously mentioned, assigning equal probabilities to each class would not have been an accurate representation of the reference. Therefore, we chose the reference to match the frequencies observed in the data.

$$RPSS = 1 - \frac{\overline{RPS_{\text{forecast}}}}{\overline{RPS_{\text{reference}}}}$$

**Equation 3: Rank Probability Skill Score (RPSS)**

As shown in equation 4, the Rank Probability Score (RPS, Weigel et al. 2007) measures the squared error between the cumulative probability mass function of a given forecast and that of a given observation. It takes on positive values, zero indicating a perfect prediction. As a result, the RPSS takes on values from  $-\infty$  to 1, where 1 indicates a perfect forecast, and 0 indicates that the model is equivalent to the reference. Negative values mean that the model does worse than the reference. Typically, for three-class classification tasks, modest predictive skill is associated with RPSS in the range [0.05, 0.20] (Goddard et al. 2003). Note that the more categories to be predicted, the harder it gets for a model to obtain high RPSS values.

$$RPS = \sum_{k=1}^K (Y_k - O_k)^2$$

**Equation 4: Rank Probability Score (RPS)**

Where  $K$  is the number of categories of the output variable,  $Y_k = \sum_{i=1}^k y_i$  is the cumulative vector of forecasted values, and  $O_k = \sum_{i=1}^k o_i$  is the cumulative vector of the observations.  $y_i$  is the probabilistic forecast for the event to happen in category  $i$ , and  $o_i=1$  if the observation is in category  $i$ , 0 else.

In order to get a feel for what the RPSS represents, consider a four-class classification problem where a probabilistic forecast for a true response  $(0, 0, 1, 0)$  is  $(0.10, 0.25, 0.60, 0.05)$ , and where the empirical frequency reference is  $(0.05, 0.15, 0.5, 0.30)$ . As shown in Tables 8 and 9, one simply needs to square the differences between the cumulative probability mass function (PMF) of the prediction given by the model and the cumulative PMF of the observation (following equation 4) to obtain the absolute skill  $RPS_{\text{forecast}}$

of the forecast. Similarly, comparing the prediction given by the reference to the observation gives the skill of the reference for that particular observation,  $RPS_{reference}$ .

Then, comparing the absolute skill of the forecast to that of the reference by computing the quantity  $1 - (RPS_{forecast}/RPS_{reference})$  gives the RPSS for the probabilistic forecast. Here,  $RPSS = 1 - \left(\frac{0.135}{0.245}\right) = 0.551$ . Note that while the model predicted the correct category with quite high confidence (60%), the RPSS is only slightly above 0.5, which illustrates well the stringency of this metric. By averaging the RPSS of many forecasts issued by a given model, one can get an accurate estimate of the predictive skill of the model.

**Table 8. Calculation of  $RPS_{forecast}$**

Category	Probabilistic Forecast	Observation	Forecast Cumulative (Y)	Observation Cumulative (O)	$(Y - O)^2$
1	0.10	0	0.1	0	0.01
2	0.25	0	0.35	0	0.1225
3	0.60	1	0.95	1	0.0025
4	0.05	0	1	1	0
$RPS_{forecast} = \text{TOTAL} = \mathbf{0.135}$					

**Table 9. Calculation of  $RPS_{reference}$**

Category	Reference	Observation	Forecast Cumulative (Y)	Observation Cumulative (O)	$(Y - O)^2$
1	0.05	0	0.05	0	0.0025
2	0.15	0	0.20	0	0.04
3	0.35	1	0.55	1	0.2025
4	0.45	0	1	1	0
$RPS_{reference} = \text{TOTAL} = \mathbf{0.245}$					

## RESULTS AND INTERPRETATION

### Predictive skill

The performance of each of the four RF and four SGTB models was evaluated by recording the RPSS for 36 runs of “leave-5%-out” cross-validation (see Figure 13). At each iteration, (1) 5% of the observations were randomly put aside without replacement from the original data set, (2) the models with the optimal parameter values determined previously were trained on the remaining 95% of observations, and finally (3) the models were tested on the 5% of left-out observations. The numbers of observations in the testing set at each round were 178, 220, 220, and 92 for the safety outcomes *body part*, *energy source*, *injury type*, and *injury severity*, respectively. These steps were repeated 36 times. The RPSS values reported in this study can, therefore, be considered highly reliable as they were computed for each model from several thousands of predictions for brand new, never seen observations. This approach provides an objective assessment of predictive skill.

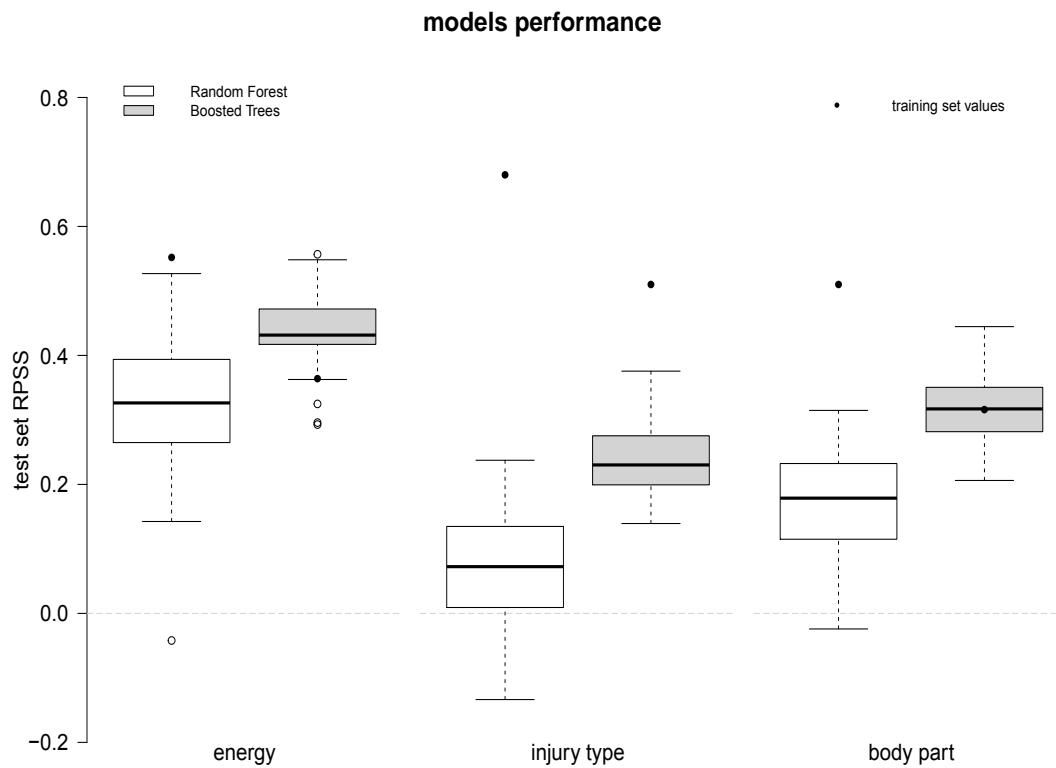
Figure 15 represents the distributions (as boxplots) of the RPSS values of the RF and SGTB models for the safety outcomes *energy type*, *injury type*, and *body part*. The thick black bars represent the means, and the circles filled in black the values on the full original data sets. The dotted horizontal line passing through the origin indicates a RPSS of zero (same skill as the reference). The mean and median RPSS values are reported in Table 11.

It can be clearly seen from Figure 15 that skill is good. More precisely, the mean RPSS values are comprised between 0.172 and 0.319 for the RF models, and between 0.236 and 0.436 for the Boosting models. This indicates medium-high to very high skill, especially considering the large number of classes to be predicted (at least 5 for each prediction task). Indeed, according to Goddard et al. (2003), modest skill is associated with RPSS values in the [0.05, 0.20] range, and very high skill is associated with RPSS values of 0.4 and above. The best performance (mean RPSS of 0.436) was attained by a SGTB model, for

the prediction of the outcome *energy type*. This represents a relative improvement of 276% over the 0.116 RPSS of the best model proposed by Esmaeili et al. (2015b), possibly justifying the choice of machine learning over parametric modeling. The high predictive skill of the models obtained can also be viewed as a proof of the validity and promising potential of the attribute-based framework. Indeed, these results show that fundamental construction attributes do carry predictive power and thus make good ML features, and that skillful and useful multi-categorical forecasts can be issued for various safety outcomes.

One can also notice from Figure 15 that all SGTB models consistently outperform their RF counterparts. This is in accordance with Caruana and Niculescu-Mizil (2006) who compared boosted trees and RF for 11 binary classification problems and found that boosted trees have a slight edge over RF (performance was evaluated based on 8 different metrics). This superiority can be partly explained by the fact that RF can only reduce error through decreasing variance (where  $\text{error} = \text{bias} + \text{variance}$ ), while Boosting reduces error on both fronts (Hastie et al. 2009, p. 588).

An example of probabilistic forecasts generated by the SGTB model for the safety outcome *injury type* (median RPSS 0.230) is provided in Table 10, along with the true response. Despite the model being the least skillful of the three SGTB models, the most likely class differs from the true response only once (marked in bold in the last column). The shades of grey indicate the magnitude of the probabilities assigned (the greater the probability, the darker).



**Figure 15. Predictive skill for the first three prediction tasks, as measured by RPSS recorded in 36 runs of cross-validation**

**Table 10. Example of probabilistic forecasts issued by the SGTB model for *injury type* (prediction error in bold)**

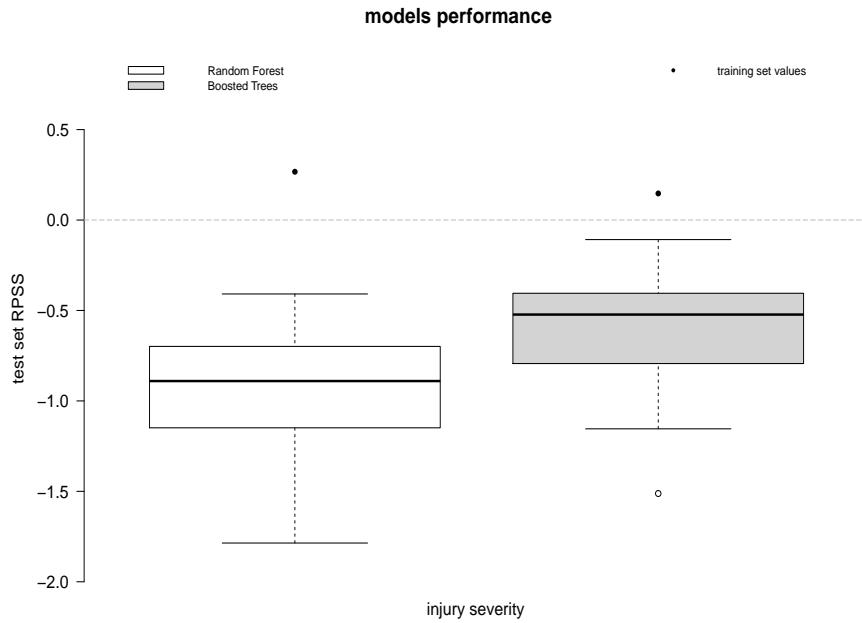
Attributes \ Outcomes	caught in or compressed	exposure to harmful sub.	fall on same level	overexertion	struck by or against	truth
hose, object on the floor	0.026	0.002	0.702	0.187	0.083	fall
ladder	0.212	0.006	0.049	0.274	0.459	<b>caught</b>
grinding, small particle	0.010	0.001	0.004	0.015	0.969	struck
concrete, formwork, heavy mat, tool, rebar, exiting/transiting	0.109	0.009	0.224	0.447	0.210	overexertion
insect	0.017	<b>0.926</b>	0.003	0.02	0.033	exposure
small particle	<b>0.0194</b>	0.001	0.005	0.0186	0.956	struck
rebar, wire, lifting pulling manual handling	0.107	0.003	0.027	0.200	0.663	struck
heat source, piping	0.055	0.863	0.005	0.031	0.047	exposure

It is interesting to note that while the reasoning behind certain predictions shown in Table 10 is clear (e.g., *heat source* → *exposure to harmful substance*, *small particle* → *struck*), in some other cases, the combinations of attributes are more complex and the most likely outcome is not as obvious or intuitive. It is in these very situations that our predictive models prove the most useful, by leveraging empirical data to guide decision-making under uncertainty.

Interestingly, as shown in Figure 16, both the RF and the SGTB model performed worse than the reference for the prediction of the fourth and last safety outcome, *injury severity*. One explanation for this absence of predictive skill is that injury severity may not be predictable simply from combinations of fundamental attributes alone. Additional predictive layers may be required, such as the amount of energy present in the environment (e.g., Alexander et al. 2015).

Also, it should be noted that a random component obviously plays a role in dictating injury severity. For instance, a worker slipping on ice may simply feel discomfort in their legs (pain), twist their ankle (first aid, medical case, or lost work time), or even badly fall backwards and sustain a head trauma (permanent disablement or fatality). Thus, in the same situation, injuries of radically different severity levels can occur based on pure chance only. Finally, injury severity as reported in accident reports is impacted by reporting practices. The same injury can be classified as pain, first aid or medical case based only on whether the injured worker chose to seek medical attention, and whether they were evaluated directly onsite or transported to some external medical facility. This injects a lot of noise.

Despite the low skill observed, the probabilistic forecast for *injury severity* could serve as a measure of *potential severity* or *potential risk of severe injury*, which can be of significant use in risk-based safety decision-making.



**Figure 16.** Predictive skill of the models for the last prediction task, as measured by RPSS recorded in 36 runs of cross-validation

**Table 11.** Mean and median RPSS for each prediction task

Prediction task:	Body part		Energy source		Injury type		Injury severity	
Model	RF	SGTB	RF	SGTB	RF	SGTB	RF	SGTB
Mean RPSS	0.172	0.324	0.319	0.436	0.068	0.236	-0.1	-0.650
Median RPSS	0.170	0.318	0.326	0.432	0.0725	0.230	-0.89	-0.522

## Variable importance

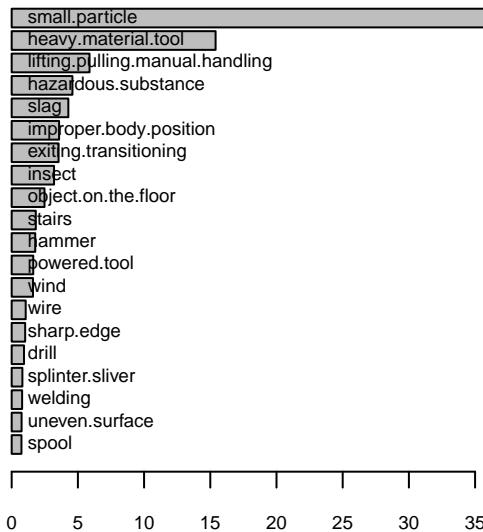
This section provides variable importance measures to attenuate the impenetrability of RF and SGTB due to their “black box” nature. Note that these values are not meaningful in absolute terms; they just allow comparison between variables. Furthermore, rather than an absolute, hard truth, the following results should be considered hints as for which fundamental attributes may play important roles in predicting the various levels of each safety outcome. Indeed, single variable importance measures only partially uncover the underlying mechanisms relating input and output variables, and are somewhat biased. For instance, as noted by Louppe (2014), variables can be assigned low importance scores not because they are uninformative, but because the signal they capture is diluted in other redundant variables. Also, predictors that look irrelevant in isolation may be relevant in combination Domingos (2012). Therefore, caution

should be used when interpreting these measures. Because they were very similar for RF and SGTB, only the rankings from the latter are provided in what follows.

### **Body part**

For the outcome *body part* (see Figure 17), the attributes *small particle*, *wind*, and *hazardous substance* (e.g., insulation) probably provide predictive power for the level *head*, as they are frequently associated with eye injuries. On the other hand, *heavy material/tool*, *lifting/pulling/manual handling*, and *improper body position* are likely to be good predictors of back injuries (i.e., the body part *trunk*). The attributes *object on the floor*, *uneven surface*, *stairs*, and *exiting/transitions* may be more related to *lower extremities* injuries (e.g., rolled ankles) caused by falls on same level. *Upper extremities* such as arms, hands and fingers are burnt by *welding* and *slag*, poked by *wires*, struck by *drills* and other *powered tools*, smashed by *hammers*, and stung/bitten by *insects*, so *upper extremities* is most likely the body part for which these precursors have the biggest influence. Finally, *insect* might be a good predictor of *neck* and *head* injuries.

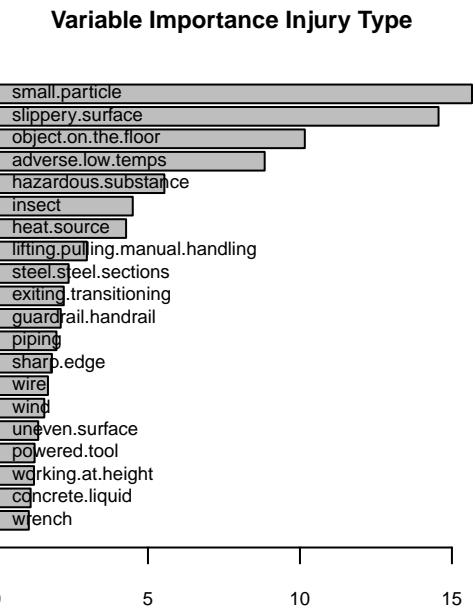
**Variable Importance Body Part**



**Figure 17. Variable importance for body part**

### **Injury type**

The variable importance measures for the outcome *injury type* are provided in Figure 18. It is probable that *steel/steel sections*, *sharp edge*, *powered tool*, *rebar*, *wire*, *small particle*, *wind*, and the compounding factor *working at height* provide predictive power for the level *struck by or against*, while *slippery surface*, *adverse low temperatures* (snow-ice), *object on the floor*, *uneven surface*, and *exiting/transitions* are associated with *fall on same level*. On the other hand, *hazardous substance*, *concrete liquid*, *insect*, *heat source*, *piping*, and also to some extent *adverse low temperatures* are plausible predictors of *exposure to harmful substance* (cold weather is considered a hazardous substance in hypothermia cases). The attributes *lifting/pulling/manual handling*, *heavy material/tool*, and *wrench* might be bringing predictive power for the category *overexertion*. *Rebar*, *powered tool*, *unpowered tool*, and *guardrail/handrail* may indicate *caught in or compressed*. Finally, *machinery* and *heavy vehicle* are probably flag *struck by or against*, *caught in or compressed*, or *exposure to harmful substance* (dangerous fluids and high temperatures) or with *fall* injuries (e.g., falls while dismounting).



**Figure 18. Variable importance for *injury type***

### ***Energy type***

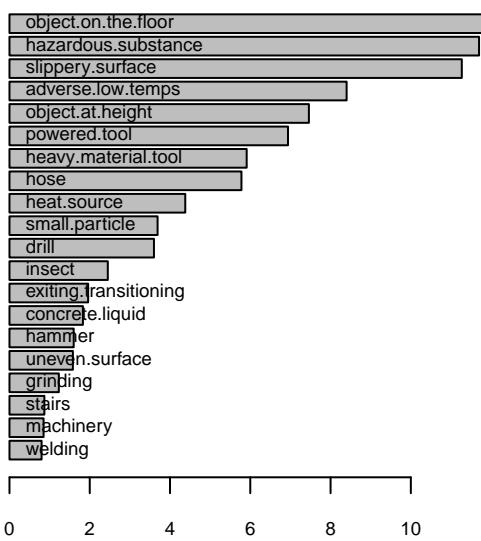
As can be inferred from Figure 19, the attributes *object on the floor*, *slippery surface*, *uneven surface*, *adverse low temperatures* and *exiting/transitioning* most likely bring substantial predicting skill for the energy level *gravity*, as these precursors are often found in cases involving slips, trips, and falls. Note that *object at height* may be also a good predictor of *gravity* injuries (of the struck by type). *Hazardous substance*, *grout*, and *concrete liquid* probably provide predictive power for the categories *chemical* and *biological*, and *insect* is obviously related to *biological*. *Heat source*, *piping*, *valve* and *welding* possibly flag reports associated with *thermal* injuries, although *welding* (like *grinding* and *drill*) may also bring substantial predictive skill for the *motion* category (flying sparks). *Adverse low temperatures* can be linked to both *gravity* injuries (as was already mentioned) and hypothermia/frostbite cases, which are classified in *thermal*. *Drill* and *powered tool* may classify reports into *mechanical* (kick-back, bit caught, etc.). The attributes *hose* and *valve* are probably bringing predictive power for the category *pressure*, but *hose* (when on the floor) can also be related to trips and falls (that is, to *gravity*-related incidents). Finally, *hammer* (finger caught), *grinding* (flying sparks), and *small particle* (flying particles) are most likely related to *motion*. The algorithms may have also learned the fact that the NLP tool developed by Tixier et al. (2016a) classifies reports into the *motion* category if they do not already fall into another category. Therefore, *motion* could be related to the absence of attributes more than to the presence of any specific attribute.

### ***Injury severity***

Finally, even though the models developed to predict injury severity exhibited low skill, it is worth noting that many attributes associated with high energy levels, such as *crane*, *heavy material/tool*, *heavy vehicle*, *drill*, and *machinery*, have high influence (see Figure 20). This tends to corroborate the preliminary findings of Alexander et al. (2015) who have shown preliminary evidence that energy magnitude predicts severity in the construction domain. It is also interesting to note that *improper procedure/inattention* and

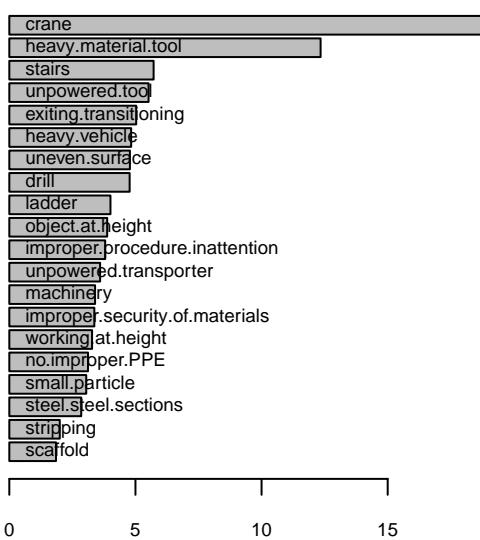
*improper security of materials* are present in the list, whereas they were not part of the most important attributes for all the previous safety outcomes. Therefore, one could infer that human factors are key players in determining injury severity, which could partly explain the difficulty of predicting this particular safety outcome.

**relative influence score**  
**Variable Importance Energy Type**



**Figure 19. Variable importance for energy type**

**relative influence score**  
**Variable Importance Injury Severity**



**Figure 20. Variable importance for injury severity**

## **CONCLUSIONS, LIMITATIONS AND RECOMMENDATIONS**

Traditional construction safety research is limited as it was built on the assumption of independence of tasks and is primarily based upon expert opinion or subjective, aggregated, or secondary data. The attribute-based framework introduced by Esmaeili and Hallowell (2012, 2011b) provided the basis for addressing both limitations, by showing possible the extraction of universal and structured safety information from raw, unstructured injury reports. However, the framework had yet to be used to its full potential due to the high cost of manual content analysis and the limitations of the statistical tools previously used for prediction. The recourse to an extended list of attributes validated by past research (Desvignes 2014, Prades Villanova 2014) and to a highly accurate NLP system (Tixier et al. 2016a) allowed a large data set of 4,400 attributes and safety outcomes to be constituted. Furthermore, we applied two state-of-the art machine learning (ML) algorithms, Random Forest (RF) and Stochastic Gradient Tree Boosting (SGTB), to this structured data set. Using binary fundamental construction attributes as input, the resulting models predict three safety outcomes out of four with high skill ( $0.236 < RPSS < 0.436$ ), namely *injury type*, *energy type*, and *body part*. This clearly outperforms the models developed in past research in terms of skill (276% relative improvement over Esmaeili et al. 2015b) but also in terms of variety of outcomes predicted. It is also to be noted that the SGTB models systematically reached higher predictive skill than their RF counterparts.

### **Contributions to theory**

The high predictive skill reached by the models for three safety outcomes out of four shows that construction injuries do not occur in a chaotic fashion, but rather that underlying patterns and trends exist and can be uncovered and captured via statistical learning when applied to sufficiently large data sets. This finding suggests that construction safety should be studied empirically like other natural phenomena rather than strictly being approached through the analysis of subjective, aggregated, or secondary data; expert-opinion; and with a regulatory and managerial perspective. Thus, this line of inquiry opens the gate

to a new research field, where construction safety is considered an empirically grounded quantitative science. The high predictive skill reached also acts as evidence that the attribute-based framework is viable, as it produces valuable structured data from unstructured injury reports. Especially, it shows that the feature engineering of Prades Villanova (2014) and Desvignes (2014) was successful. It also justifies the choice of algorithmic modeling over parametric modeling.

The absence of skill for the output variable *injury severity* suggests that unlike other safety outcomes, *injury severity* is mainly random, or that additional layers of predictive information should be taken into account in making predictions. Examples of such information may include the energy level in the environment (e.g., Alexander et al. 2015). Future research should try to incorporate such energy-based data into the predictive models to test whether predictive skill can be improved for *injury severity*. However, large-scale gathering of this information remains a challenge as it does not seem to be accessible from text. Nonetheless, current predictions for *injury severity* can be used as an estimate of *potential injury severity risk*.

Also, it should be noted that current predictions are conditional on the occurrence of an accident. Indeed, all that can be learned from attribute and outcome data extracted from injury reports is what happens when an accident occurs. Making unconditional predictions would necessitate the recording of “non-accident” cases. Such data, currently unavailable, could be gathered by making random observations of the conditions onsite in terms of attributes.

Other suggestions for future research include extracting attributes and outcomes from larger amounts of injury reports, in order to overcome the absolute rarity issue faced in this research for certain levels of the target variables. This should yield improvement in predictive skill for all prediction tasks. Also, using training data extracted from injury reports originating from other sectors than the industrial, energy, infrastructure, and mining ones would widen the range of application of the models. Another way to

improve the current predictions would be to train a learning algorithm that combines the predictions of various models: RF, SGTB, but also others, such as support vector machines or artificial neural networks. This approach, known in the ML field as model *stacking*, has proven highly successful (Domingos 2012).

To sum up, this study makes important strides in that the final models can provide reliable probabilistic forecasts of the likely outcomes should an accident occur in a given construction situation. This kind of predictions had been absent from the field since its inception. Safety analysts in the broader context may also find important methodological advancements in the extraction of structured data from unstructured text via NLP and the attribute-based framework, and from subsequent prediction made via ML. This combination opens the field to automated safety analysis from massive data sets (i.e., “big data”).

### **Contributions to practice**

Professionals have long aimed to add prediction to safety. The field of construction safety research has recently grown to include risk analysis, leading indicators, and precursor analysis. To achieve the goal of being predictive, practitioners have turned to expert input, particularly from knowledgeable safety professionals. However, as human beings, even the most experimented safety experts have limited personal history with injuries (thousands of worker hours), and a plethora of cognitive biases alter their judgment under uncertainty. On the other hand, the ML algorithms used in this study learned lessons from large volumes of objective, empirical data corresponding to millions of worker hours.

This objective knowledge can be used to complement potentially biased individual opinions, leading to better-informed, safer decision-making. For example, a user simply needs to identify the attributes expected for a work package and the new models can predict, with good accuracy, the type of energy, type of injury, and body part involved should an accident occur. Such actionable feedback can be used to better plan a worksite by removing (in time and/or space), replacing, or communicating attributes before

exposure. Also, the predictions can be used to better target pre-job safety meetings. For example, a forecasted high probability of hand injury can be used to spur focused discussions about proper gloves for the task, or the prediction of a high probability for the pressure type of energy can encourage focusing hazard recognition programs on sources of pressure energy.

Finally, these predictions have great potential for integration with advanced work packaging and building information modeling software as the models use binary attributes as input variables. Before construction work begins, designers, engineers, and planners can be provided with predictions of the most likely outcomes should an accident occur. Also, new configurations can be considered and objectively balanced against time, cost, and quality as a competing criterion. Safety professionals have long languished the fact that safety is considered as a fragmented function. The attribute-based framework of Esmaeili and Hallowell (2012, 2011b), coupled with the NLP tool of Tixier et al. (2016a) and with the methodology proposed in this study may take strides toward true, objective integration of empirical safety data within construction planning and design.

**CHAPTER 5: CONSTRUCTION SAFETY RISK MODELING AND  
SIMULATION**

## **ABSTRACT**

By building on a recently introduced genetic-inspired attribute-based conceptual framework for safety risk analysis, we propose a novel methodology to compute construction univariate and bivariate construction safety risk at a situational level. Our fully data-driven approach provides construction practitioners and academicians with an easy and automated way of extracting valuable empirical insights from databases of unstructured textual injury reports. By applying our methodology on an attribute and outcome dataset directly obtained from 814 injury reports, we show that the frequency-magnitude distribution of construction safety risk is very similar to that of natural phenomena such as precipitation or earthquakes. Motivated by this observation, and drawing on state-of-the-art techniques in hydroclimatology and insurance, we introduce univariate and bivariate nonparametric stochastic safety risk generators, based on Kernel Density Estimators and Copulas. These generators enable the user to produce large numbers of synthetic safety risk values faithfully to the original data, allowing safety-related decision-making under uncertainty to be grounded on extensive empirical evidence. Just like the accurate modeling and simulation of natural phenomena such as wind or streamflow is indispensable to successful structure dimensioning or water reservoir management, we posit that improving construction safety calls for the accurate modeling, simulation, and assessment of safety risk. The underlying assumption is that like natural phenomena, construction safety may benefit from being studied in an empirical and quantitative way rather than qualitatively which is the current industry standard. Finally, a side but interesting finding is that attributes related to high energy levels (e.g., machinery, hazardous substance) and to human error (e.g., improper security of tools) emerge as strong risk shapers on the dataset we used to illustrate our methodology.

## **INTRODUCTION AND MOTIVATION**

Despite the significant improvements in safety that have followed the inception of the Occupational Safety and Health Act of 1970, safety performance has reached a plateau in recent years and construction

still accounts for a disproportionate accident rate. From 2013 to 2014, fatalities in construction even increased by 5% to reach 885, the highest count since 2008 (Bureau of Labor Statistics 2015). In addition to terrible human costs, construction injuries are also associated with huge direct and indirect economic impacts.

Partly due to their limited personal history with accidents, even the most experienced workers and safety managers may miss hazards and underestimate the risk of a given construction situation (Albert et al. 2014, Carter and Smith 2006). Designers face an even greater risk of failing to recognize hazards and misestimating risk (Albert et al. 2014, Almén and Larsson 2012). Therefore, a very large portion of construction work, upstream or downstream of ground-breaking, involves safety-related decision-making under uncertainty. Unfortunately, even more when uncertainty is involved, humans often recourse to personal opinion and intuition to apprehend their environment. This process is fraught with numerous biases and misconceptions inherent to human cognition (e.g., Kahneman and Tversky 1982) and compounds the likelihood of misdiagnosing the riskiness of a situation.

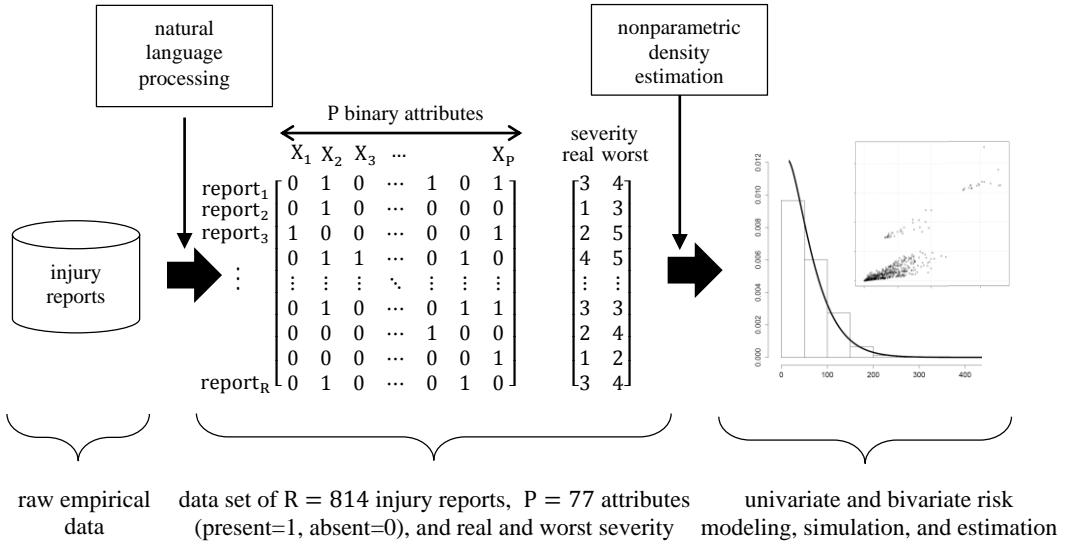
Therefore, it is of paramount importance to provide construction practitioners with tools to mitigate the adverse consequences of uncertainty on their safety-related decisions. In this study, we focus on leveraging situational data extracted from raw textual injury reports to guide and improve construction situation risk assessment. Our methodology facilitates the augmentation of construction personnel's experience and grounds risk assessment on potentially unlimited amounts of empirical and objective data. Put differently, our approach combats construction risk misdiagnosis on two fronts, by jointly addressing both the limited personal history and the judgment bias problems previously evoked.

We leveraged attribute data extracted by a highly accurate Natural Language Processing (NLP) system (Tixier et al. 2016a) from a database of 921 injury reports provided by a partner organization engaged in industrial construction projects worldwide.

Fundamental construction attributes are context-free universal descriptors of the work environment. They are observable prior to injury occurrence and relate to environmental conditions, construction means and methods, and human factors. To illustrate, one can extract four attributes from the following text: "worker is unloading a ladder from pickup truck with bad posture": ladder, manual handling, light vehicle, and improper body positioning. Because attributes can be used as leading indicators of construction safety performance (Tixier et al. 2016b, Esmaeili et al. 2015b), they are also called injury precursors. In what follows, we will use the terms attribute and precursor interchangeably.

Drawing from national databases, Esmaeili and Hallowell (2012, 2011) initially identified 14 and 34 fundamental attributes from 105 fall and 300 struck-by high severity injury cases, respectively. In this study we used a refined and broadened list of 80 attributes carefully engineered and validated by Prades Villanova (2014) and Desvignes (2014) from analyzing a large database of 2,201 reports featuring all injury types and severity levels. These attributes, along with their counts and final risk values in our dataset, are summarized in Table 1. Note that as will be explained later, risk values are unitless and do not have physical meaning. They are only meaningful in that they allow comparison between attributes.

A total of 107 out of 921 reports were discarded because they were not associated with any attribute and because the real outcome was unknown, respectively. Additionally, 3 attributes out of 80 (pontoon, soffit, and poor housekeeping) were removed because they did not appear in any report. This gave a final matrix of  $R = 814$  reports by  $P = 77$  attributes. While other related studies concerned themselves with pattern recognition and predictive modeling (e.g., Chapters 2 and 3 of the present dissertation, Esmaeili et al. 2015b), here we focus on construction safety risk analysis. The study pipeline is summarized in Figure 1.



**Figure 1. Overarching research process: from raw injury reports to safety risk analysis**

The contributions of this study are fourfold: (1) we formulate an empirically-grounded definition of construction safety risk at the attribute level, and extend it to the situational level, both in the univariate and the bivariate case; (2) we show how to model risk using Kernel density estimators; (3) we observe that the frequency-magnitude distribution of risk is heavy-tailed, and resembles that of many natural phenomena; and finally, (4) we introduce univariate and bivariate nonparametric stochastic generators based on Kernels and Copulas to draw conclusions from much larger samples and better estimate construction safety risk.

## BACKGROUND AND POINT OF DEPARTURE

To understand how the present study departs from and contributes to the current body of knowledge, we present in what follows a broad review of the safety risk analysis literature. Traditional risk analysis methods for construction safety are limited in two major ways: in terms of the (1) data used (primarily opinion-based), and in terms of the (2) level of analysis (typically trade, activity or task).

**Table 1. Relative risks and counts of the P = 77 injury precursors**

precursor	n	e (%)	risk based on		precursor	n	e (%)	risk based on				
			worst possible					worst possible				
			real	possible				real	possible			
concrete	29	41	7	96	unstable support/surface	3	32	1	2			
confined workspace	21	2	115	336	wind	29	37	6	16			
crane	16	12	22	76	improper body position	7	25	3	6			
door	17	21	11	174	imp. procedure/inattention	13	16	10	44			
sharp edge	8	38	2	5	imp. security of materials	78	12	77	1007			
formwork	22	5	63	135	insect	19	18	8	21			
grinding	16	16	11	34	no/improper PPE	3	67	0*	1			
heat source	11	20	4	13	object on the floor	41	43	9	22			
heavy material/tool	29	30	11	247	lifting/pulling/handling	141	31	49	439			
heavy vehicle	12	12	12	307	cable tray	9	27	4	11			
ladder	23	14	15	52	cable	8	33	1	3			
light vehicle	31	59	7	123	chipping	4	16	1	4			
lumber	69	14	53	158	concrete liquid	8	41	2	4			
machinery	40	8	67	3159	conduit	11	31	4	14			
manlift	8	8	16	50	congested workspace	2	32	0*	1			
object at height	14	50	4	136	dunnage	2	16	1	3			
piping	74	38	19	141	grout	3	41	1	1			
scaffold	91	33	28	74	guardrail handrail	16	40	4	8			
stairs	28	41	8	25	job trailer	2	59	0*	1			
steel/steel sections	112	35	33	281	stud	4	41	1	5			
rebar	33	4	76	251	spool	9	33	2	9			
unpowered transporter	13	9	23	401	stripping	12	22	7	18			
valve	24	27	9	22	tank	16	31	5	115			
welding	25	22	10	34	drill	16	43	5	88			
wire	30	43	5	19	bolt	36	41	7	27			
working at height	73	40	18	46	cleaning	22	56	5	12			
wkg below elev. wksp/mat.	7	17	3	21	hammer	33	50	5	18			
forklift	11	9	9	380	hose	11	41	3	8			
hand size pieces	38	47	7	95	nail	15	50	4	10			
hazardous substance	33	1	590	6648	screw	7	50	1	2			
adverse low temps	33	3	101	292	slag	10	10	8	32			
mud	6	6	9	20	spark	1	12	2	11			
poor visibility	3	23	2	3	wrench	23	39	5	23			
powered tool	32	27	12	54	exiting/transitions	25	49	6	17			
slippery surface	32	25	13	40	splinter/sliver	9	44	1	2			
small particle	96	31	28	105	working overhead	5	40	1	3			
unpowered tool	102	44	24	352	repetitive motion	2	51	0*	1			
electricity	1	33	0*	1	imp. security of tools	24	22	12	314			
uneven surface	33	32	11	129								

\* values are rounded up to the nearest integer

## Data

While the data used differ widely among construction safety risk studies, three main sources emerge from the literature: expert opinion, government statistics, and empirical data obtained from construction organizations or national databases. The vast majority of studies use opinion-based data, and thus rely on the ability of experts to rate the relative magnitude of risk based on their professional experience. Often, ranges are provided by researchers to bound risk values. Additionally, even the most experienced experts

have limited personal history with hazardous situations, and their judgment under uncertainty suffer the same cognitive limitations as that of any other human being (Capen 1976). Some of these judgmental biases include overconfidence, anchoring, availability, representativeness, unrecognized limits, motivation, and conservatism (Rose 1987, Tversky and Kahneman 1981, Capen 1976). It has also been suggested that gender and even current emotional state have an impact on risk perception (Tixier et al. 2014, Gustafsod 1998). Even if it is possible to somewhat alleviate the negative impact of adverse psychological factors (e.g., Hallowell and Gambatese 2009b), the reliability of data obtained from expert opinion is questionable. Conversely, truly objective empirical data, like the injury reports used in this study, seem superior.

### **Level of analysis**

Due to the technological and organizational complexity of construction work, most safety risk studies assume that construction processes can be decomposed into smaller parts (Lingard 2013). Such decomposition allows researchers to model risk for a variety of units of analysis. For example, Hallowell and Gambatese (2009a), Navon and Kolton (2006), and Huang and Hinze (2003) focused on specific tasks and activities. Most commonly, trade-level risk analysis has been adopted (Baradan and Usman 2006, Jannadi and Almishari 2003, Everett 1999). The major limitation of these segmented approaches is that because each one considers a trade, task, or activity in isolation, it is impossible for the user to comprehensively characterize onsite risk in a standard, robust and consistent way.

Some studies attempted to address the aforementioned limitations. For instance, Shapira and Lyachin (2009) quantified risks for very generic factors related to tower cranes such as type of load or visibility, thereby allowing safety risk modeling for any crane situation. Esmaeili and Hallowell (2012, 2011) went a step further by introducing a novel conceptual framework allowing *any* construction situation to be fully and objectively described by a unique combination of fundamental context-free attributes of the work

environment. This attribute-based approach is powerful in that it shows possible the extraction of structured standard information from naturally occurring, unstructured textual injury reports. Additionally, the universality of attributes allows to capture the multifactorial nature of safety risk in the same unified way for any task, trade, or activity, which is a significant improvement over traditional segmented studies. However, manual content analysis of reports is expensive and fraught with data consistency issues. For this reason, Tixier et al. (2016a) introduced a Natural Language Processing (NLP) system capable of automatically detecting the attributes presented in Table 1 and various safety outcomes in injury reports with more than 95% accuracy (comparable to human performance), enabling the large scale use of the attribute-based framework. The data we used in this study was extracted by the aforementioned NLP tool.

## UNIVARIATE ANALYSIS

### Attribute-level safety risk

Following Baradan and Usmen (2006), we defined construction safety risk as the product of frequency and severity as shown in equation 1. More precisely, in our approach, the safety risk  $R_p$  accounted for by precursor<sub>p</sub> (or  $X_p$  in Tables 1 and 2) was computed as the product of the number  $n_{ps}$  of injuries attributed to precursor<sub>p</sub> for the severity level s (given by Table 2) and the impact rating  $S_s$  of this severity level (given by Table 3, and based on Hallowell and Gambatese 2009a). We considered five severity levels,  $s_1 = \text{Pain}$ ,  $s_2 = \text{First Aid}$ ,  $s_3 = \text{Medical Case/Lost Work Time}$ ,  $s_4 = \text{Permanent Disablement}$ , and  $s_5 = \text{Fatality}$ . Medical Case and Lost Work Time were merged because differentiating between these two severity levels turned out to be challenging based on the information available in the narratives only.

$$\text{risk} = \text{frequency} \cdot \text{severity}$$

**Equation 1. Construction safety risk**

**Table 2. Counts of injury severity levels accounted for by each precursor**

Precursors	Severity levels				
	$s_1 = \text{Pain}$	$s_2 = \text{1st Aid}$	$s_3 = \text{Medical Case/Lost Work Time}$	$s_4 = \text{Permanent Disablement}$	$s_5 = \text{Fatality}$
$X_1$	$n_{11}$	$n_{12}$	$n_{13}$	$n_{14}$	$n_{15}$
$X_2$	$n_{21}$	$n_{22}$	$n_{23}$	$n_{24}$	$n_{25}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$X_{P-1}$	$n_{(P-1)1}$	$n_{(P-1)2}$	$n_{(P-1)3}$	$n_{(P-1)4}$	$n_{(P-1)5}$
$X_P$	$n_{P1}$	$n_{P2}$	$n_{P3}$	$n_{P4}$	$n_{P5}$

**Table 3. Severity level impact scores**

Severity Level ( $s$ )	Severity scores ( $S_s$ )
Pain	$S_1 = 12$
1 <sup>st</sup> Aid	$S_2 = 48$
Medical Case/Lost Work Time	$S_3 = 192$
Permanent Disablement	$S_4 = 1024$
Fatality	$S_5 = 26214$

The total amount of risk that can be attributed to precursor<sub>p</sub> was then obtained by summing the risk values attributed to this precursor across all severity levels, as shown in equation 2.

$$R_p = \sum_{s=1}^5 (n_{ps} \cdot S_s)$$

**Equation 2. Total amount of risk associated with precursor<sub>p</sub>**

Where  $n_{ps}$  is the number of injuries of severity level  $s$  attributed to precursor<sub>p</sub>, and  $S_s$  is the impact score of severity level  $s$

Finally, as noted by Sacks et al. (2009), risk analysis is inadequate if the likelihood of worker exposure to specific hazards is not taken into account. Hence, the risk  $R_p$  of precursor<sub>p</sub> was weighted by its probability of occurrence  $e_p$  onsite (see equation 3), which gave the relative risk  $RR_p$  of precursor<sub>p</sub>. The

probabilities  $e_p$ , or exposure values, were provided by the same company that donated the injury reports. These data are constantly being recorded by means of observation as part of the firm's project control and work characterization policy, and therefore were already available.

$$RR_p = \frac{1}{e_p} \cdot R_p = \frac{1}{e_p} \cdot \sum_{s=1}^5 (n_{ps} \cdot S_s)$$

**Equation 3. Relative risk for precursor<sub>p</sub>**

Where  $R_p$  is the total amount of risk associated with precursor<sub>p</sub>, and  $e_p$  is the probability of occurrence of precursor<sub>p</sub> onsite.

To illustrate the notion of relative risk, assume that the precursor lumber has caused 15 first aid injuries, 10 medical cases and lost work time injuries, and has once caused a permanent disablement. By following the steps outlined above, the total amount of risk  $R_{lumber}$  accounted for by the attribute lumber can be computed as  $15 \times 48 + 10 \times 192 + 1 \times 1024 = 3664$ . Moreover, if lumber is encountered frequently onsite, e.g., with an exposure value  $e_{lumber} = 0.65$ , the relative risk of lumber will be  $RR_{lumber} = 3664/0.65 = 5637$ . On the other hand, if workers are very seldom exposed to lumber (e.g.,  $e_{lumber} = 0.07$ ),  $RR_{lumber}$  will be equal to  $3664/0.07 = 52343$ . It is clear from this example that if two attributes have the same total risk value, the attribute having the lowest exposure value will be associated with the greatest relative risk. The assumption is that if a rare attribute causes as much damage as a more common one, the rare attribute should be considered riskier. Note that relative risk values allow comparison but do not have an absolute physical meaning. As presented later, what matters more than the precise risk value itself is the range in which the value falls.

Also, note that since Tixier et al.'s (2016a) NLP tool's functionality did not include injury severity extraction at the time of writing, we used the real and worst possible outcomes manually assessed for each

report by Prades Villanova (2014). Specifically, in Prades Villanova (2014), a team of 7 researchers analyzed a large database of injury reports over the course of several weeks. High output quality was ensured by using a harsh 95% inter-coder agreement threshold, peer-reviews, calibration meetings, and random verifications by an external reviewer. Regarding worst possible injury severity, human coders were asked to use their judgment of what would have happened in the worst case scenario should a small translation in time and/or space had occurred. This method and the resulting judgments were later validated by Alexander et al. (2015) who showed that the human assessment of maximum possible severity was congruent with the quantity of energy in the situation, which, ultimately, is a reliable predictor of the worst possible outcome.

For instance, in the following excerpt of an injury report: “worker was welding below scaffold and a hammer fell from two levels above and scratched his arm”, the real severity is a first aid. However, by making only a small translation in space, the hammer could have struck the worker in the head, which could have yielded a permanent disablement or even a fatality. Furthermore, coders were asked to favor the most conservative choice; that is, here, permanent disablement. Whenever mental projection was impossible or required some degree of speculation, coders were required to leave the field as blank and the reports were subsequently discarded. As indicated, Alexander et al. (2015) empirically validated these subjective assessments.

By considering severity counts for both real outcomes and worst possible outcomes, we could compute two relative risk values for each of the 77 precursors. These values are listed in Table 1, and were stored in two vectors of length  $P = 77$ .

For each attribute, we computed the difference between the relative risk based on worst possible outcomes and the relative risk based on actual outcomes. The top 10% attributes for this metric, which can be considered the attributes that have the greatest potential for severity escalation should things go

wrong, are hazardous substance ( $\Delta = 6059$ ), machinery (3092), improper security of materials (930), lifting/pulling/manual handling (390), unpowered transporter (378), forklift (371), unpowered tool (328), improper security of tools (302), and heavy vehicle (295). Except lifting/pulling/manual handling and unpowered tool, all these precursors are directly associated with human error or high energy levels, which corroborates recent findings (Tixier 2015, Alexander et al. 2015, respectively). Furthermore, one could argue that the attributes lifting/pulling/manual handling and unpowered tool are still indirectly related to human error and high energy levels, as the former is often associated with improper body positioning (human factor) while the latter usually designates small and hand held objects (hammer, wrench, screwdriver, etc.) that are prone to falling from height (high energy). Many attributes in Table 1, such as sharp edge, manlift, unstable support/surface, or improper body position, have low risk values because of their rarity in the rather small data set that we used to illustrate our methodology, but this does not incur any loss of generality.

### **Report-level safety risk**

By multiplying the (R, P) attribute binary matrix (attribute matrix of Figure 1) by each (P, 1) relative risk vector (real and worst) as shown in equation 4, two risk values were obtained for each of the R = 814 incident reports. This operation was equivalent to summing the risk values based on real and worst possible outcomes of all the attributes that were identified as present in each report (see equation 5).

For instance, in the following description of a construction situation: “worker is unloading a ladder from pickup truck with bad posture”, four attributes are present: namely (1) *ladder*, (2) *manual handling*, (3) *light vehicle*, and (4) *improper body positioning*. The risk based on real outcomes for this construction situation can be computed as the sum of the relative risk values of the four attributes present (given by Table 1), that is,  $15 + 49 + 7 + 3 = 74$ , and similarly, the risk based on worst potential outcomes can be computed as  $52 + 439 + 123 + 6 = 620$ .

$$\begin{array}{c}
 \text{P precursors} \\
 \overbrace{\left[ \begin{array}{cccccc} 0 & 1 & 0 & \cdots & 1 & 0 & 1 \\ 0 & 1 & 0 & \cdots & 1 & 0 & 0 \\ 1 & 0 & 0 & \cdots & 0 & 0 & 1 \\ 0 & 1 & 1 & \cdots & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 1 & 0 & \cdots & 0 & 1 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 1 \\ 0 & 1 & 0 & \cdots & 0 & 1 & 0 \end{array} \right]}_{(R, P)} & \cdot & \left[ \begin{array}{c} RR_1 \\ RR_2 \\ RR_3 \\ \vdots \\ RR_{(P-2)} \\ RR_{(P-1)} \\ RR_P \end{array} \right]_{(P, 1)} & = & \left[ \begin{array}{c} R_{\text{report}_1} \\ R_{\text{report}_2} \\ R_{\text{report}_3} \\ R_{\text{report}_4} \\ \vdots \\ R_{\text{report}_{(R-3)}} \\ R_{\text{report}_{(R-2)}} \\ R_{\text{report}_{(R-1)}} \\ R_{\text{report}_R} \end{array} \right]_{(R, 1)}
 \end{array}$$

**Equation 4. Safety risk at the report level (a)**

Multiplying the  $(R, P)$  attribute matrix by the  $(P, 1)$  vector of relative risk values for each attribute gives the  $(R, 1)$  vector of risk values associated with each injury report.

$$R_{\text{report}_r} = \sum_{p=1}^P (RR_p \cdot \delta_{rp})$$

**Equation 5. Safety risk at the report level (b)**

Where  $RR_p$  is the relative risk associated with precursor<sub>p</sub>, and  $\delta_{rp} = 1$  if precursor<sub>p</sub> is present in report<sub>r</sub> ( $\delta_{rp} = 0$  else).

As already stressed, these relative values are not meaningful in absolute terms, they only enable comparison between situations and their categorization into broad ranges of riskiness (e.g., low, medium, high). Estimating these ranges on a small, finite sample such as the one we used in this study would have resulted in biased estimates. To alleviate this, we used stochastic simulation techniques to generate hundreds of thousands of new scenarios honoring the historical data, enabling us to make inferences from a much richer, yet faithful sample.

### The probability distribution of construction safety risk resembles that of many natural phenomena

For a given injury report, the risk based on real outcomes and the risk based on worst potential outcomes can each take on a quasi-infinite number of values ( $2^P - 1$ ) with some associated probabilities.

Therefore, they can be considered quasi-continuous random variables, and have legitimate probability distribution functions (PDFs). Furthermore, since a risk value cannot be negative by definition, these PDFs have  $[0, +\infty)$  support.

The empirical PDF of the risk based on real outcomes for the 814 injury reports is shown as a histogram in Figure 2. The histogram divides the sample space into a number of intervals and simply counts how many observations fall into each range. We can clearly see that the empirical safety risk is right-skewed and exhibits a thick tail feature. In other words, the bulk of construction situations present risk values in the small-medium range, while only a few construction situations are associated with high and extreme risk. This makes intuitive sense and is in accordance with what we observe onsite, i.e., frequent benign injuries, and low-frequency high-impact accidents.

Such heavy-tailed distributions are referred to as “power laws” in the literature, after Pareto (1896), who proposed that the relative number of individuals with an annual income larger than a certain threshold was proportional to a power of this threshold. Power laws are ubiquitous in nature (Pinto et al. 2012, Malamud 2004). Some examples of natural phenomena whose magnitude follow power laws include earthquakes, ocean waves, volcanic eruptions, asteroid impacts, tornadoes, forest fires, floods, solar flares, landslides, and rainfall (Papalexiou et al. 2013, Pinto et al. 2012, Menéndez et al. 2008, Malamud et al. 2006). Other human related examples include insurance losses and healthcare expenditures (Ahn et al. 2012), hurricane damage cost (Jagger et al. 2008, Katz 2002), and the size of human settlements and files transferred on the web (Reed 2001, Crovella and Bestavros 1995).

To highlight the resemblance between construction safety risk and some of the aforementioned natural phenomena, we selected four datasets that are standard in the field of extreme value analysis, and freely available from the “extRemes” R package (Gilleland and Katz 2011). We overlaid the corresponding

PDFs with that of construction safety risk. For ease of comparison, variables were first rescaled as shown in equation 6. In what follows, each data set is briefly presented.

$$Z = \frac{X - \min(X)}{\max(X) - \min(X)}$$

#### **Equation 6. Variable rescaling**

Where X is the variable in the original space and Z is the variable in the rescaled space.

#### *Summer maximum temperatures in Arizona*

The first dataset reported summer maximum temperatures in Phoenix, AZ, from 1948 to 1990, measured at Sky harbor airport. The observations were multiplied by -1 (flipped horizontally) before rescaling. The distribution is named “max temperature” in Figure 3.

#### *Hurricane economic damage*

The second dataset (“hurricane damage” in Figure 3) consisted in total economic damage caused by every hurricane making landfall in the United States between 1925 and 1995, expressed in 1995 U.S. \$ billion. Following Katz’s (2002) recommendation, all individual storms costing less than \$0.01 billion were removed to minimize potential biases in the recording process. The final number of hurricanes taken into account was 86.

#### *Potomac River peak flow*

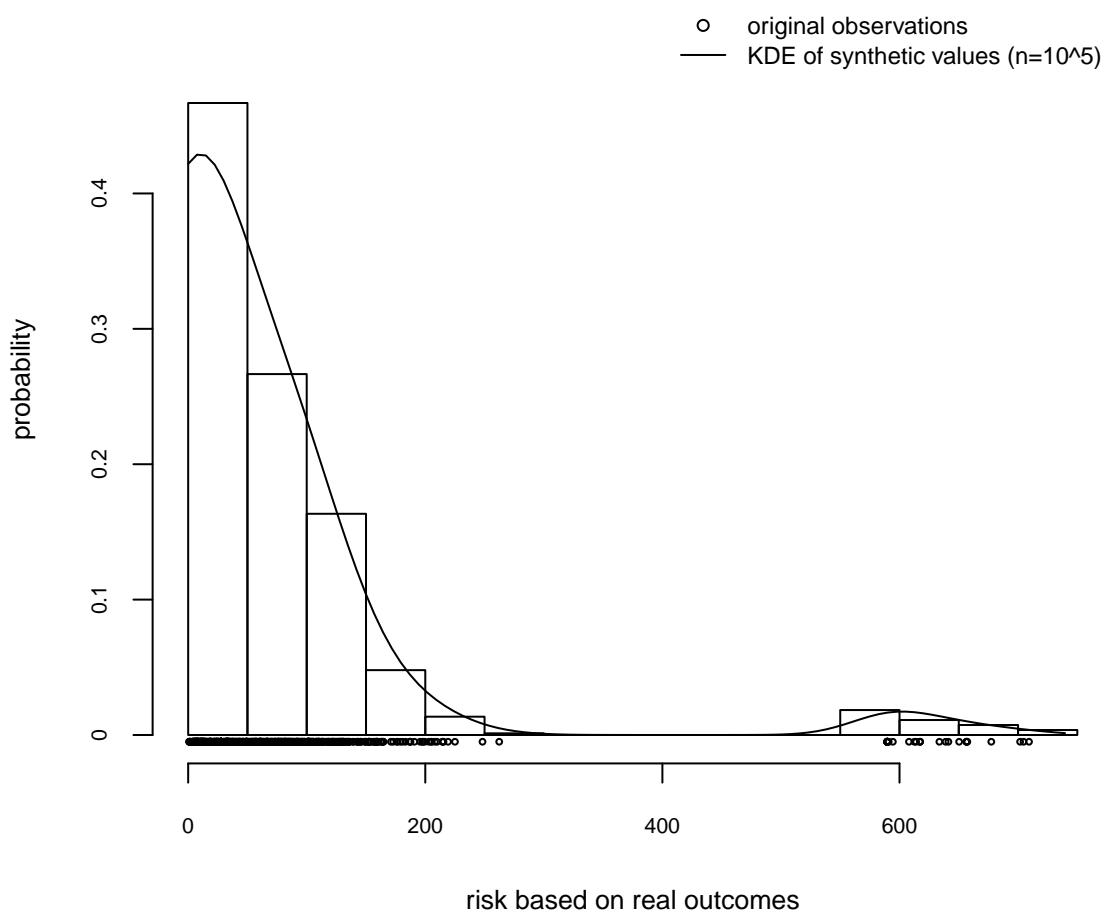
The third data set included in our comparison was observations of Potomac River peak stream flow measured in cubic feet per second at Point Rocks, MD, from 1895 to 2000. The observations were divided by  $10^5$  before rescaling. The curve is labeled “peak flow” in Figure 3.

### *Precipitation in Fort Collins, CO*

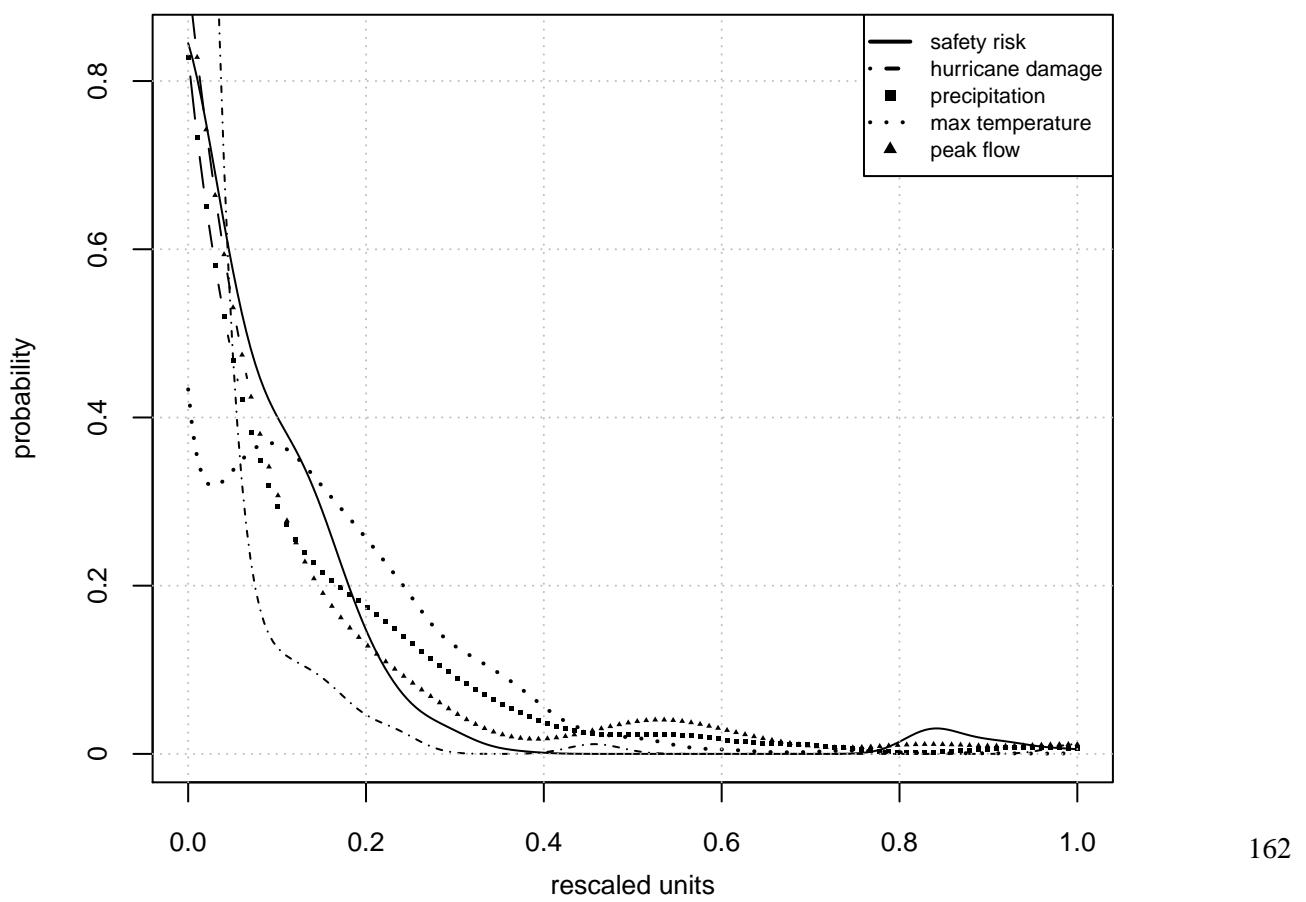
The fourth and last dataset contained 36,524 daily precipitation amounts (in inches) from a single rain gauge in Fort Collins, CO. Only values greater than 1 inch were taken into account, giving a final number of 213 observations. The distribution is named “precipitation” in Figure 3.

We estimated the PDFs by using kernel density estimators (KDE) since overlaying histograms would have resulted in an incomprehensible figure. The KDE, sometimes called Parzen, is a nonparametric way to estimate a PDF. It can be viewed as a smoothed version of the histogram, where a continuous function, called the Kernel, is used rather than a box as the fundamental constituent (Silvermann 1986, p. 3). The Kernel has zero mean, is symmetric, positive, and integrates to one. The last two properties ensure that the Kernel, and as a result the KDE, is a probability distribution. More precisely, as shown in equation 7, the KDE at each point  $x$  is the average contribution from each of the Kernels at that point (Hastie et al. 2009 p. 208). Put differently, the KDE at  $x$  is a local average of functions assigning weights to the neighboring observations  $x_i$  that decrease as  $|x_i - x|$  increases (Saporta 2011, p. 323, Moon et al. 1995). The “local” estimation is the key feature of this method in enabling to capture the features present in the data. KDEs converge faster to the underlying density than the histogram, and are robust to the choice of the origin of the intervals (Moon et al. 1995).

**Figure 2. Histogram of original observations (n=814) with boundary corrected KDE of the simulated observations (n=10<sup>5</sup>)**



**Figure 3. Comparison of safety risk with natural phenomena**



$$\hat{f}_X(x) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}, h\right)$$

**Equation 7. Kernel Density Estimator (KDE)**

Where  $\{x_1, \dots, x_n\}$  are the observations,  $K$  is the Kernel, and  $h$  is a parameter called the bandwidth. Note that  $\hat{f}_X$  is an estimator of the true PDF  $f_X$ , which is unknown.

$h$  is a parameter called the bandwidth that controls smoothing and therefore affects the final shape of the estimate (Hastie et al. 2009, p. 193). A large bandwidth creates a great amount of smoothing, which decreases variance and increases bias as the fit to the observations is loose. In that case, most of the structure in the data is not captured (i.e., underfitting). On the other hand, a small bandwidth will tightly fit the data and its spurious features such as noise (i.e., overfitting), which yields a low bias but a high variance. There is definitely a tradeoff here. In this study, we used a standard and widespread way of estimating  $h$  called Silverman's rule of thumb (Silverman 1986 p. 48) shown in Equation 8. We invite the reader to reference Rajagopalan et al. (1997a) for a good review of the objective bandwidth selection methods.

$$h = \frac{0.9 \min\left(\hat{\sigma}_X, \frac{Q_3 - Q_1}{1.34}\right)}{n^{1/5}}$$

**Equation 8. Silverman's rule of thumb for bandwidth selection**

Where  $Q_3$  and  $Q_1$  are the third and first quartiles (respectively),  $\hat{\sigma}_X$  is the standard deviation of the sample, and  $n$  is the size of the sample. Here,  $n = R = 814$ .

Further, for our Kernel  $K$ , we selected the standard Normal distribution  $N(0,1)$ , that is, the Normal distribution centered on zero with unit variance. Since the PDF of  $N(0,1)$  is  $\frac{1}{\sqrt{2\pi}} e^{-x^2/2}$ , the associated KDE can be written using Equation 7 as shown in Equation 9. Other popular Kernels include the

triangular, biweight or Epanechnikov, but the consensus in the statistics literature is that the choice of the Kernel is secondary to the estimation of the bandwidth (e.g., Saporta 2011, p. 323).

$$\hat{f}_X(x) = \frac{1}{nh\sqrt{2\pi}} \sum_{i=1}^n e^{-\frac{1}{2}\left(\frac{x_i-x}{h}\right)^2}$$

#### **Equation 9. KDE with standard Normal Kernel**

Where  $\{x_1, \dots, x_n\}$  are the observations, and  $h$  is the bandwidth. Here,  $n = R = 814$ .

It is well known that the KDE suffers a bias at the edges on bounded supports. Indeed, because the Kernel functions are symmetric, weights are assigned to values outside the support, which causes the density near the edges to be significantly underestimated, and creates a faulty visual representation. In our case, safety risk takes on values in  $[0, +\infty)$ , so issues arise when approaching zero. We used the correction for the boundary bias via local linear regression (Jones 1993) using the “evmix” package (Hu and Scarott 2014) of the R programming language (R core team 2015). Boundary reflection and log transformation are other popular approaches for controlling boundary bias (Rajagopalan et al. 1997a, Silverman 1986).

#### **Why does construction safety risk follow a power law?**

The power law behavior of construction safety risk can be explained from a technical standpoint by the “inverse of quantities” mechanism. As Newman (2005) explains, any quantity  $X \sim Y^{-\gamma}$  for a given  $\gamma$  will have a probability distribution  $P[X] \sim X^{-\alpha}$ , with  $\alpha = 1 + 1/\gamma$ . Further, it can be shown that this probability distribution exhibits power law behavior.

In the special case of construction safety risk, by simply using the fact that  $RR_p = \frac{1}{e_p} \cdot R_p$  (equation 3), we can rewrite equation 5 as equation 10.

$$R_{\text{report}_r} = \sum_{p=1}^P \left( \frac{1}{e_p} \cdot R_p \cdot \delta_{rp} \right)$$

**Equation 10. Risk at the report level (c)**

Where  $e_p$  is the probability of occurrence of precursor<sub>p</sub> onsite,  $R_p$  is the total amount of risk associated with precursor<sub>p</sub>, and  $\delta_{rp} = 1$  if precursor<sub>p</sub> is present in report<sub>r</sub> (0 else).

Finally, setting  $X = R_{\text{report}_r}$  and  $Y = \prod_1^P e_p$ , it follows from equation 10 that  $X \sim Y^{-\gamma}$  with  $\gamma = 1$ , which, according to Newman (2005), suffices to show that  $R_{\text{report}_r}$  is power law distributed. Further, Newman (2005) stresses that even though the relationship between X and Y is already some sort of power law (X is proportional to a power of Y), this relationship is deterministic, not stochastic. Still, it generates a power law probability distribution, which is not trivial.

Moreover, the large values of  $R_{\text{report}_r}$ , those in the tail of the distribution, correspond to large values of  $RR_p$ , that is, to small values of  $e_p$  close to zero (i.e., rare precursors). This makes sense, and is in accordance with the theory of extremes (extreme values are rare).

There are more underlying processes that can generate fat tails in the distributions of natural and other human-related phenomena, such as multiplicative processes (Adlouni et al. 2008, Mitzenmacher 2004) random walks, the Yule process, self-organized criticality, and more (Newman 2005). They cannot be all addressed here. Moreover, the *inverse of quantities* mechanism seems to be the most plausible and most straightforward explanation for the shape of the probability distribution of construction safety risk observed in this study.

## Univariate modeling

In this section, we focus on construction safety risk based on real outcomes. We present a computational method that can be used to generate synthetic safety risk values that honor the historical data. Note that while many techniques and concepts in risk modeling and management deal with extreme values only, in this study we seek to capture and simulate from the entire risk spectrum (not only the extremes) in order to accurately assess the safety risk of any construction situation.

Today, extreme value analysis is still a subject of active research, and is widely used in a variety of different fields. In addition to the modeling of extreme hydroclimatological events, its applications include insurance losses (Guillen et al. 2011) and financial market shock modeling (Glantz and Kissell 2014). A central quantity in risk management is the quantile.

The quantile function (or simply quantile, for short) of a continuous random variable  $X$  is defined as the inverse of its cumulative distribution function (CDF) as shown in equation 11. The CDF is obtained by integrating or summing the PDF, respectively in the continuous and discrete case.

$$Q(p) = F_X^{-1}(p)$$

### Equation 11. Quantile function

Where  $F_X$  is the CDF of  $X$  defined as  $F_X(x) = P[X \leq x] = p \in [0,1]$

The quantile is closely linked to the concept of exceedances. In finance and insurance for instance, the value-at-risk for a given horizon is the loss that cannot be exceeded with a certain probability of confidence within the time period considered, which is given by the quantile function. For instance, the 99.95% value-at-risk  $Q(99.95)$  at 10 days represents the amount of money that the loss can only exceed

with 0.5% probability in the next 10 days. In other words, the corresponding fund reserve would cover 199 losses over 200 ( $199/200=0.995$ ).

The quantile function is also associated with the notion of return period  $T$  in hydroclimatology. For example, the magnitude of the 100-year flood ( $T = 100$ ) corresponds to the streamflow value that is only exceeded by 1% of the observations, assuming one observation per year. This value is given by  $Q(1 - 1/T) = Q(0.99)$ , which is the 99<sup>th</sup> percentile, or the 99<sup>th</sup> 100-quantile. Similarly, the magnitude of the 500-year flood,  $Q(0.998)$ , is only exceeded by 0.2% of the observations. For construction safety, this quantity would correspond to the minimum risk value that is only observed on average in one construction situation over five hundred. The median value, given by  $Q(0.5)$ , would correspond to the safety risk observed on average in one construction situation over two.

### **Limitations of traditional parametric techniques**

Traditional approaches to quantile estimation are based on parametric models of PDF especially from the Extreme Value Theory (EVT) framework (Coles et al. 2001). These models possess fat tails unlike traditional PDFs, and thus are suitable for robust estimation of extremes. The candidate distributions from the EVT are Frechet, Weibull, Gumbel, GEV, Generalized Pareto, or mixtures of these distributions (Charpentier and Oulidi 2010). These parametric models are powerful in that they allow complex phenomena to be entirely described by a single mathematical equation and a few parameters. However, being parametric, these models tend to be suboptimal when little knowledge is available about the phenomenon studied (which is the case in this exploratory study). Indeed, even when enough data are available and all parameters are estimated accurately, conclusions may be irrelevant in the case of initial model misspecification (Charpentier and Oudini 2010, Charpentier et al. 2007, Breiman 2001a). This is very problematic, especially when risk-based decisions are to be made from these conclusions.

In addition, parametric models, even from the EVT, are often too lightly tailed to avoid underestimating the extreme quantiles (Vrac and naveau 2007), which is a major limitation as accurately capturing the tail of a probability distribution is precisely the crucial thing in risk management (Frigessi et al. 2002). A popular remediation strategy consists in fitting a parametric model to the tail only, such as the Generalized Pareto, but selecting a threshold that defines the tail is a highly subjective task (Scarrott and MacDonald 2012), and medium and small values, which represent the bulk of the data are overlooked (Vrac and Naveau 2007). What is clearly better, however, especially when the final goal is simulation, is to capture the entire distribution. As a solution, hydroclimatologists have proposed dynamic mixtures of distributions, based on weighing the contributions of two overlapping models, one targeting the bulk of the observations, and the other orientated towards capturing extremes (Furrer and Katz 2007, Frigessi et al. 2002). Unfortunately, threshold selection implicitly carries over through the estimation of the parameters of the mixing function, and with most mixing functions, conflicts arise between the two distributions around the boundary (Hu and Scarrott 2013). For all these reasons, we decided to adopt a fully data-driven, nonparametric approach that we describe below.

### **Univariate construction safety risk generator**

The proposed approach consists in generating independent realizations from the nonparametric PDF estimated via the KDE described above. We base our generator on the smoothed bootstrap with variance correction proposed by Silverman (1986, p. 142-145). Unlike the traditional nonparametric bootstrap (Efron 1979) that simply consists in resampling with replacement, the smoothed bootstrap can generate values outside of the historical limited range, and does not reproduce spurious features of the original data such as noise (Rajagopalan et al. 1997b). The smoothed bootstrap approach has been successfully used in modeling daily precipitation (Lall et al., 1996), streamflow (Sharma et al., 1997) and daily weather (Rajagopalan et al., 1997b).

More precisely, the algorithm that we implemented in R to generate our synthetic values can be broken down into the following steps:

For  $j$  in 1 to the desired number of simulated values:

1. choose  $i$  uniformly with replacement from  $\{1, \dots, R\}$
2. sample  $\epsilon_X$  from the standard normal distribution with variance  $h_X^2$
3. record  $X_{sim,j} = \bar{X} + (X_i - \bar{X} + \epsilon_X) / \sqrt{1 + h_X^2 / \hat{\sigma}_X^2}$

Where  $R = 814$  is the sample size (the number of injury reports),  $\bar{X}$  and  $\hat{\sigma}_X^2$  are the sample mean and variance, and  $h_X^2$  is the variance of the standard normal Kernel (bandwidth of the KDE). Note that we deleted the negative simulated values to be consistent with the definition of risk.

Figure 2 shows the KDE of the  $10^5$  simulated values overlaid with the histogram of the original sample. It can be clearly seen that the synthetic values are faithful to the original sample since the PDF from the simulated values fit the observations very well. Also, while honoring the historical data, the smoothed bootstrap generated values outside the original limited range, as desired. The maximum risk value in our sample was 709, while the maximum of the simulated values was 740 (rounded to the nearest integer). Table 4 compares the quantile estimated via the quantile() R function of the original and simulated observations.

**Table 4. Quantile estimates based on original and simulated values for the risk based on real outcomes**

	safety risk observed in one situation over:						
	2	5	10	100	500	1,000	10,000
Original observations (n = R = 814)	57	110	152	649	703	706	709
Simulated observations (n = $10^5$ )	61	116	154	647	700	708	728

The quantile estimates of Table 4 are roughly equivalent before reaching the tails. This is because the bulk of the original observations were in the low to medium range, enabling quite accurate quantile estimates for this range in the first place. The problem stemmed from the sparsity of the high to extreme values in the historical sample, which made estimation of the extreme quantiles biased. Our use of the smoothed bootstrap populated the tail space with new observations, yielding a slightly higher estimate of the extreme quantiles, as can be seen in Table 4. It makes sense that the extremes are higher than what could have been inferred based simply on the original sample, as the original sample can be seen as a finite window in time whereas our simulated values correspond to observations that would have been made over a much longer period. The chance of observing extreme events is of course greater over a longer period of time.

Based on estimating the quantiles on the extended time frame represented by the synthetic values, we propose the risk ranges shown in Table 5. As already explained, these ranges are more robust and unbiased as the ones that would have been built from our historical observations. Thanks to this empirical way of assessing safety risk, construction practitioners will be able to adopt an optimal proactive approach by taking coherent preventive actions and provisioning the right amounts of resources.

**Table 5. Proposed ranges for the risk based on real outcomes**

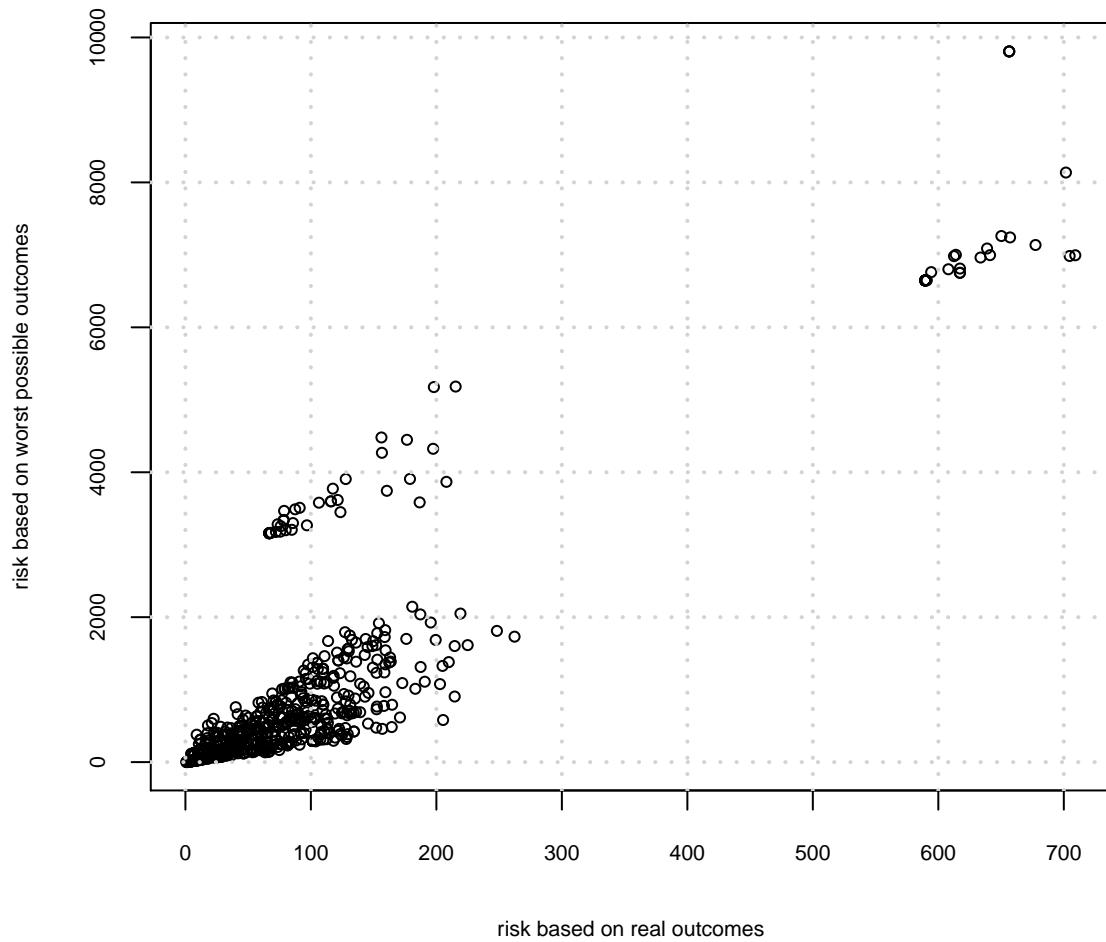
quantiles	0	0.25	0.50	0.75	0.99	1
risk value	0	29	61	105	647	740
range	<i>low</i>	<i>medium</i>	<i>high</i>	<i>very high</i>	<i>extreme</i>	

## BIVARIATE ANALYSIS

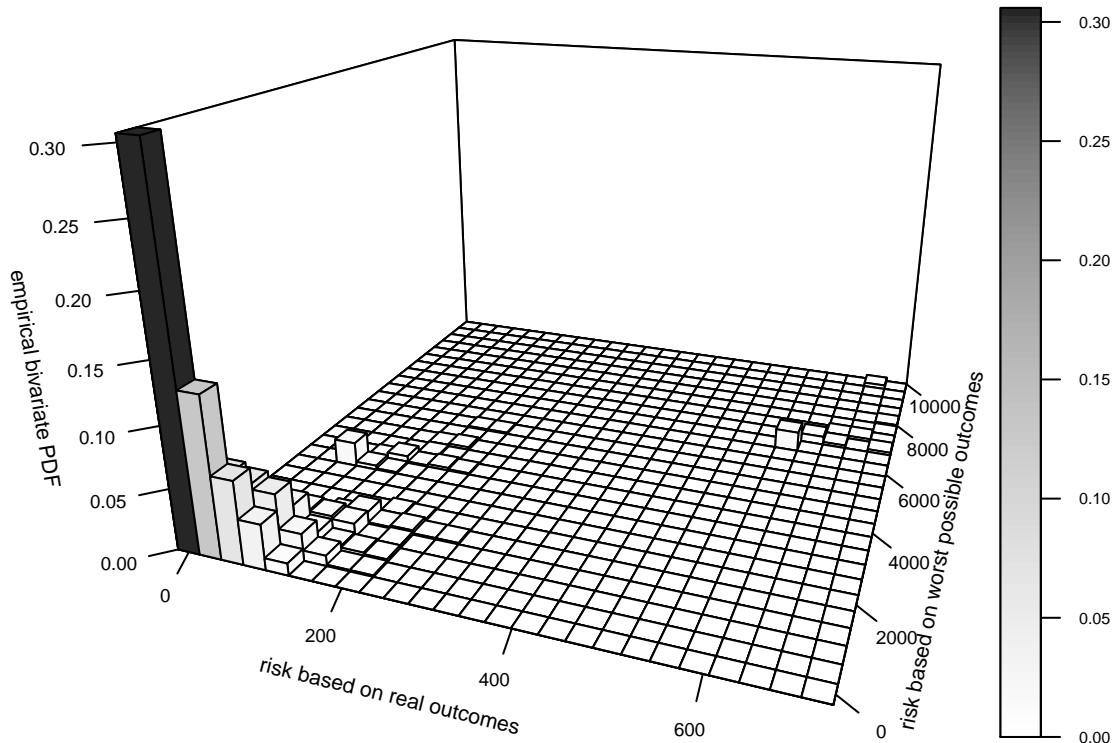
In what follows, we study the relationship between the risk based on real outcomes (X, for brevity) and the risk based on worst potential outcomes (Y). Rather than assuming that these variables are independent and considering them in separation, we acknowledge their dependence and aim at capturing it, and fatefully reproducing it in our simulation engine. This serves the final goal of being able to accurately assess the potential of an observed construction situation for safety risk escalation should the worst case scenario occur. Figure 4 shows a plot of Y versus X, while a bivariate histogram can be seen in Figure 5.

We can distinguish three distinct regimes in Figure 4. The first regime, corresponding roughly to  $0 < X < 70$ , is that of benign situations that stay benign in the worst case. Under this regime, there is limited potential for risk escalation. The second regime ( $70 < X < 300$ ) shows that beyond a certain threshold, moderately risky situations can give birth to hazardous situations in the worst case. The attribute responsible for the switch into this second regime is machinery (e.g., welding machine, generator, pump). The last regime ( $X > 300$ ) is that of the extremes, and features clear and strong upper tail dependence. The situations belonging to this regime are hazardous in their essence and create severe outcomes in the worst case scenarios. In other words, those situations are dangerous in the first place and unforgiving. The attribute responsible for this extreme regime is hazardous substance (e.g., corrosives, adhesives, flammables, asphyxiants). Again, note that these examples are provided as a result of applying our methodology on a data set of 814 injury reports for illustration purposes but do not incur any loss of generality. Using other, larger data sets would allow risk regimes to be characterized by different and possibly more complex attribute patterns.

**Figure 4. Bivariate construction safety risk**



**Figure 5. Bivariate histogram with 25 by 25 grid**



## Bivariate modeling

Many natural and human-related phenomena are multifactorial in essence and as such their study requires the joint modeling of several random variables. Traditional approaches consist in modeling their dependence with the classical family of multivariate distributions, which is clearly limiting, as it requires all variables to be separately characterized by the same univariate distributions (called the margins). Using Copula theory addresses this limitation by creating a joint probability distribution for two or more variables while preserving their original margins (Hull 2006). In addition to the extra flexibility they offer, the many existing parametric Copula models are also attractive in that they can model the dependence among a potentially very large set of random variables in a parsimonious manner (i.e., with only a few parameters). For an overview of Copulas, one may refer to Cherubini et al. (2004).

While the introduction of Copulas can be tracked back as early as 1959 with the work of Sklar, they did not gain popularity until the end of the 1990s when they became widely used in finance. Copulas are now indispensable to stochastic dependence problem understanding (Durante et al. 2010), and are used in various fields from cosmology to medicine. Since many hydroclimatological phenomena are multidimensional, Copulae are also increasingly used in hydrology, weather and climate research, for instance for precipitation infilling, drought modeling, and extreme storm tide modeling (Bárdossy et al. 2014, Domino et al. 2014, Salvadori et al. 2007).

Formally, a d-dimensional Copula is a joint CDF with  $[0,1]^d$  support and standard uniform margins (Charpentier 2006). Another equivalent definition is given by Sklar's (1959) theorem, which states in the bivariate case that the joint CDF  $F(x,y)$  of any pair  $(X,Y)$  of continuous random variables can be written as in Equation 12.

$$F(x, y) = C\{F_X(x), F_Y(y)\}, \quad (x, y) \in \mathbb{R}^2$$

**Equation 12. Sklar's theorem**

Where  $F_X$  and  $F_Y$  are the respective margins of  $X$  and  $Y$ , and  $C: [0,1]^2 \rightarrow [0,1]$  is a Copula.

Note that equation 12 is consistent with the first definition given, because for any continuous random variable  $X$  of CDF  $F_X$ ,  $F_X(X)$  follows a uniform distribution.

However, parametric Copulas suffer from all the limitations inherent to parametric modeling briefly evoked previously. Therefore, like in the univariate case, we decided to use a fully data-driven, nonparametric approach to Copula density estimation. We used the bivariate KDE to estimate the empirical Copula, which is defined as the joint CDF of the rank-transformed (or pseudo) observations.

The pseudo-observations are obtained as shown in Equation 13.

$$U_X(x) = \frac{\text{rank}(x)}{\text{length}(X) + 1}$$

**Equation 13. Rank-transformation.**

Where  $U_X$  is the transformed sample of the pseudo observations,

and  $X$  is the original sample.

Because the Copula support is the unit square  $[0,1]^2$ , the KDE boundary issue arises twice this time, near zero and one, yielding multiplicative biases (Charpentier et al. 2007). Therefore, the density is even more severely underestimated than in the univariate case, and it is even more crucial to ensure robustness of the KDE at the corners to ensure proper visualization. For this purpose, we used the transformation trick described by Charpentier et al. (2007) as our boundary correction technique. The original idea was proposed by Devroye and Györfi (1985). The approach consists in using a transformation  $T$  bijective, strictly increasing, continuously differentiable, and which has a continuously differentiable inverse, such

that  $X' = T(X)$  is unbounded. A KDE can therefore be used to estimate the density of  $X'$  without worrying about boundary bias. Finally, a density estimate of  $X$  can be obtained via back-transformation, as shown in equation 14.

$$\hat{f}_X(x) = \frac{\hat{f}_{X'}(x')}{\left| \frac{d}{dx'} T^{-1}(x') \right|} \Bigg|_{x'=T(x)}$$

**Equation 14. Transformation trick**

Where  $\hat{f}_X$  is the boundary-corrected KDE of  $X$ ,  $\hat{f}_{X'}$  is the KDE of  $X'$ , and  $T$  is the transformation such that

$$X' = T(X)$$

We used the inverse CDF of the Normal distribution,  $F_{N(0,1)}^{-1}$  as our transformation  $T$ . It goes from  $[0,1]$  to the real line. The resulting empirical Copula density estimate of the original sample is shown in Figure 6.

### Bivariate construction safety risk generator

Like in the univariate case, we used a nonparametric, fully data driven approach, the *smoothed bootstrap with variance correction*, as our simulation scheme. Minor adaptations were needed due to the two-dimensional nature of the task. The steps of the algorithm that we implemented using the R programming language are outlined below, and the resulting  $10^5$  simulated values are shown in Figure 7. Note that the procedure is equivalent to simulating from the nonparametric Copula density estimate introduced above. Like in the univariate case, we deleted the negative simulated values to ensure consistency with the definition of risk.

For j in 1 to the desired number of simulated values:

1. choose i uniformly with replacement from  $\{1, \dots, R\}$
2. sample  $\epsilon_X$  from the standard normal distribution with variance  $h_X^2$ , and  $\epsilon_Y$  from the standard normal distribution with variance  $h_Y^2$
3. take:

$$X_{simj} = \bar{X} + (X_i - \bar{X} + \epsilon_X) / \sqrt{1 + h_X^2 / \hat{\sigma}_X^2}$$

$$Y_{simj} = \bar{Y} + (Y_i - \bar{Y} + \epsilon_Y) / \sqrt{1 + h_Y^2 / \hat{\sigma}_Y^2}$$

4. record:

$$U_{simj} = F_{N(0,1)}(X_{simj}), V_{simj} = F_{N(0,1)}(Y_{simj})$$

Where  $R = 814$  is the number of injury reports,  $\bar{X}$  and  $\hat{\sigma}_X^2$  are the mean and variance of X;  $\bar{Y}$  and  $\hat{\sigma}_Y^2$  are the mean and variance of Y;  $h_X^2$  is the bandwidth of the KDE of X;  $h_Y^2$  is the bandwidth of the KDE of Y; and  $F_{N(0,1)}$  is the CDF of the standard Normal distribution, the inverse of our transformation T.

Note that step 1 selects a pair  $(x,y)$  of values from the original sample  $(X,Y)$ , not two values independently. This is crucial in ensuring that the dependence structure is preserved. Step 4 sends the simulated pair to the pseudo space to enable visual comparison with the empirical Copula density estimate, which is defined in the unit square (i.e., rank space). We can clearly observe in Figure 7 that our sampling scheme was successful in generating values that reproduce the structure present in the original data, validating our nonparametric approach. For the sake of completeness, we also compared (see Figures 8 and 9) the simulated pairs in the original space with the original values. Once again, it is easy to see that the synthetic values honor the historical data. To enable comparison with the univariate case (see Table 4), Table 6 summarizes the empirical quantiles for the historical and simulated observations of risk

based on worst potential outcomes (i.e.,  $Y$ ). Like in the univariate case, we can observe that the differences between the estimates increase with the quantiles. Notably, simulation allows to obtain richer estimates of the extreme quantiles,  $Q(1 - \frac{1}{1000}) = Q(0.999)$  and  $Q(1 - \frac{1}{10000}) = Q(0.9999)$ , whereas with the initial limited sample, the values of the quantile function plateau after  $Q(1 - \frac{1}{500}) = Q(0.998)$  due to data sparsity in the tail. Similarly to Table 5, we also propose in Table 7 ranges for the risk based on worst potential outcomes.

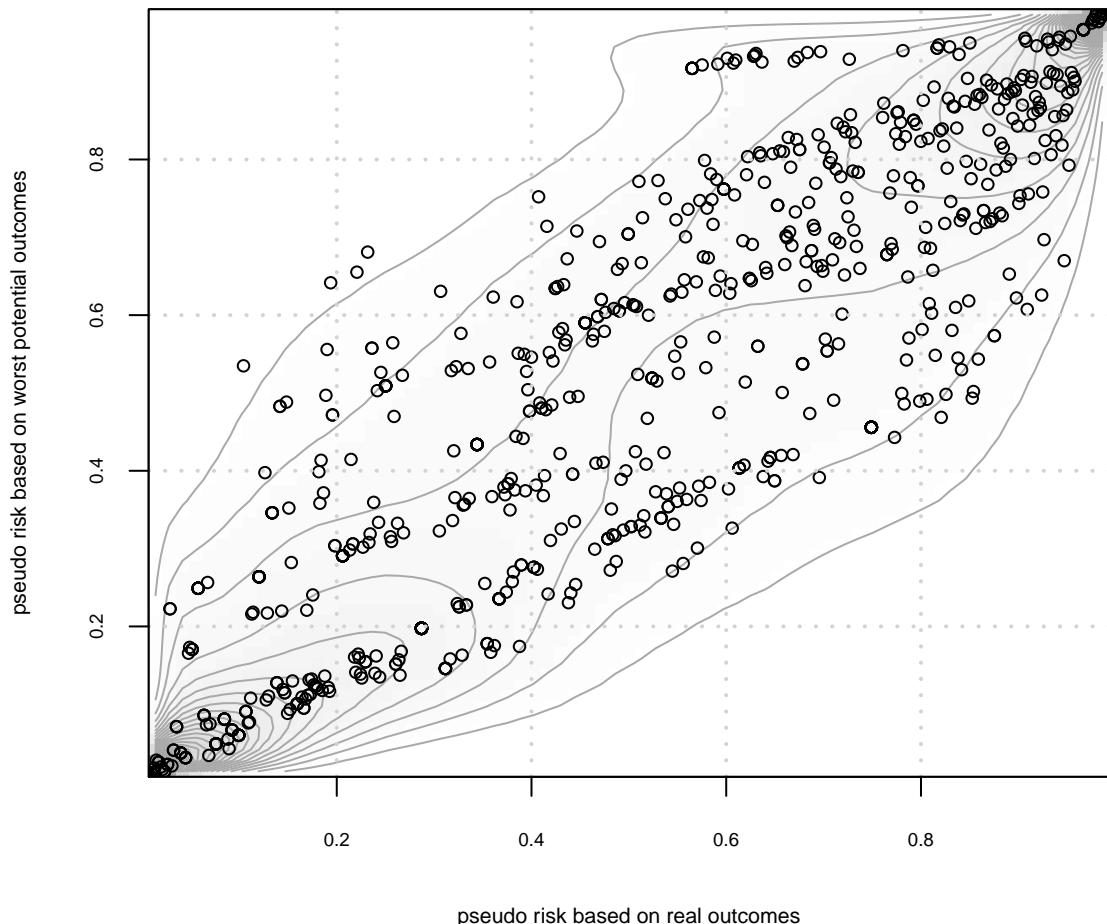
**Table 6. Quantile estimates based on original and simulated values for the risk based on worst potential outcomes**

safety risk observed in one situation over:							
	2	5	10	100	500	1,000	10,000
original observations (n = R = 814)	343	950	1719	7000	9808	9808	9808
simulated observations (n = 10 <sup>5</sup> )	395	1061	1953	7092	9765	9586	10045

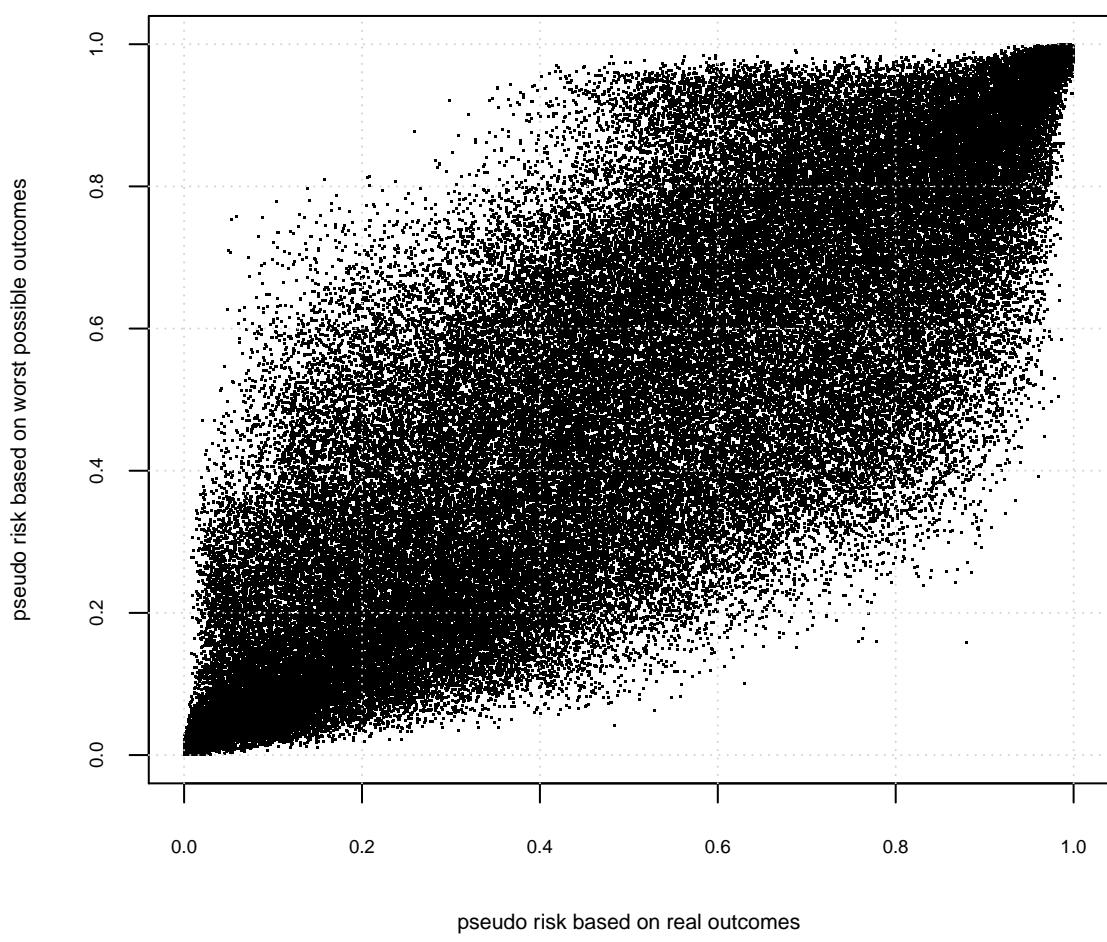
**Table 7. Proposed ranges for the risk based on worst potential outcomes**

quantiles	0	0.25	0.50	0.75	0.99	1
risk value	0	183	395	837	7092	10126
range	<i>low</i>	<i>medium</i>	<i>high</i>	<i>very high</i>	<i>extreme</i>	

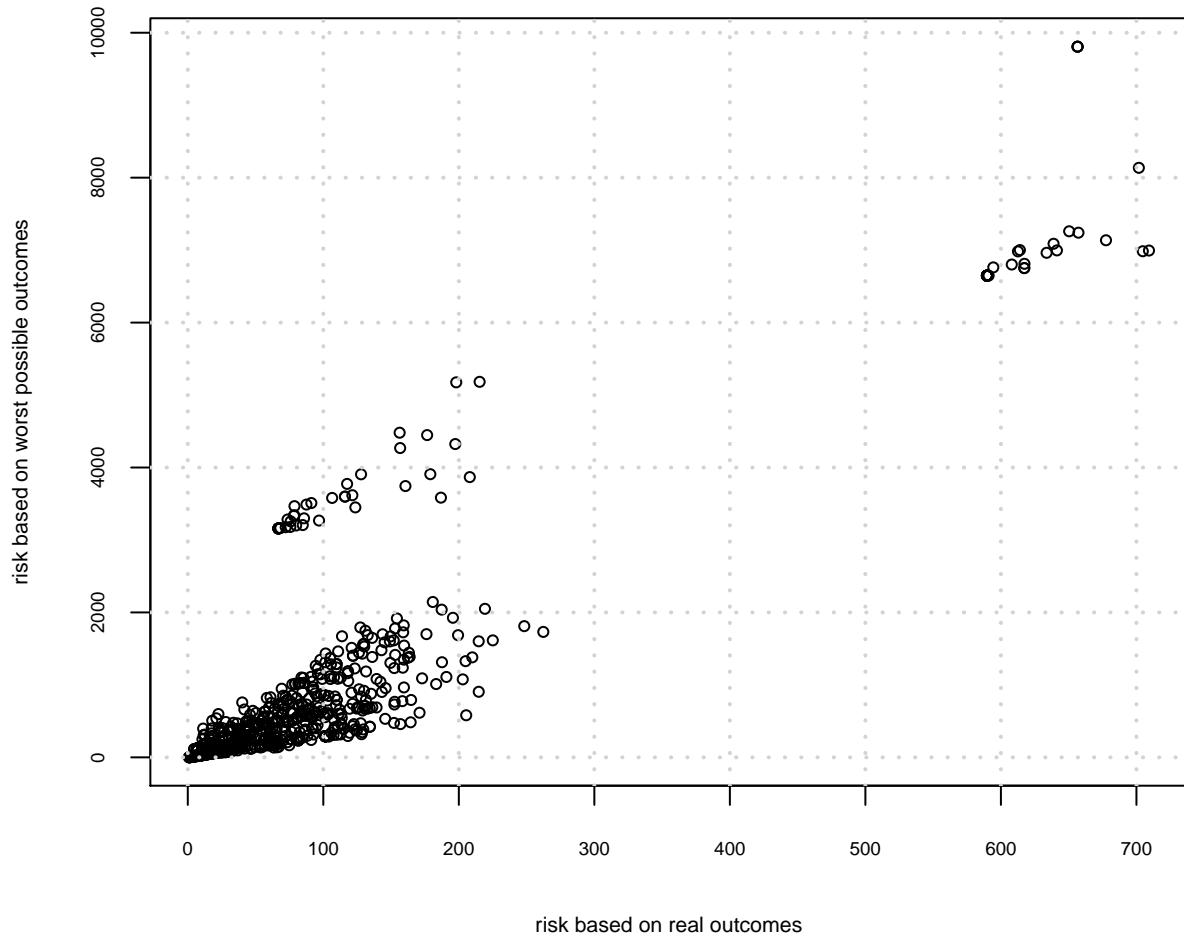
**Figure 6. Nonparametric Copula density estimate  
with original pseudo-observations**



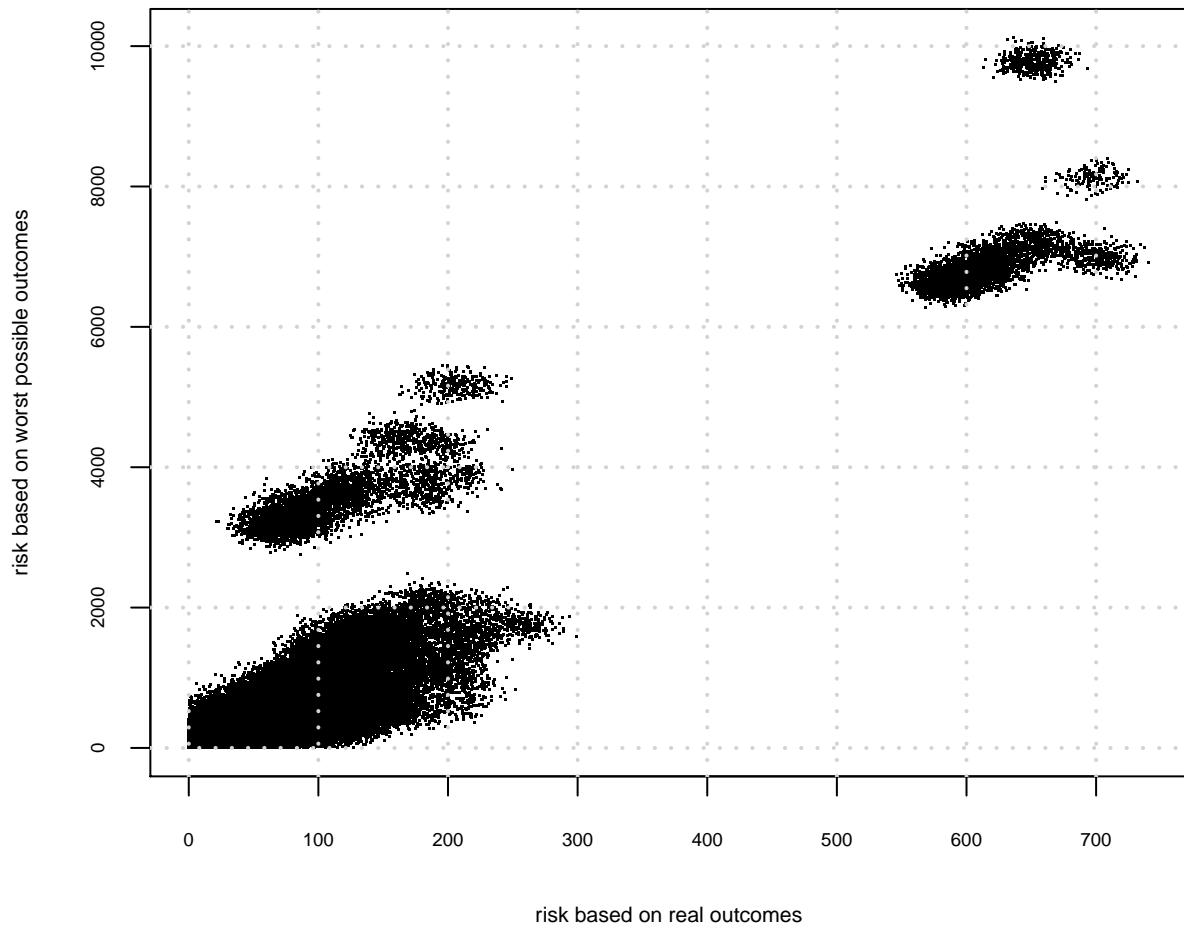
**Figure 7. Simulated risk values in rank space  
 $n=10^5$**



**Figure 8. Bivariate construction safety risk**



**Figure 9. simulated risk values in original space  
n=10<sup>5</sup>**



### **Computing risk escalation potential based on simulated values**

Using the synthetic safety risk pairs obtained via our bivariate stochastic safety risk generator, and evidence provided by the user (i.e., an observation made onsite in terms of attributes), it is possible to compute and estimate of the upper limit of risk, i.e., the safety risk presented by the observed construction situation based on worst case scenarios. This estimate is based on large numbers of values simulated in a data-driven approach that features the same dependence structure as the original, empirical data. The end user (e.g., designer or a safety manager) can therefore make data-based, informed decisions, and proactively implement the adequate remediation strategies. Furthermore, the attribute-based nature of the procedure is ideally suited for automated integration with building information modeling and work packaging. The technique we propose, based on conditional quantile estimation, consists in the steps detailed in what follows.

First, the attributes observed in a particular construction situation give the risk based on real outcomes for the construction situation, say  $x_0$ . By fixing the value of X to  $x_0$ , this first step extracts a slice from the empirical bivariate distribution of the simulated values. This slice corresponds to the empirical probability distribution of Y conditional on the value of X, also noted  $P[Y|X = x_0]$ . Because only a few values of Y may exactly be associated with  $x_0$ , we consider all the values of Y associated with the values of X in a small neighboring range around  $x_0$ , that is,  $P[Y|x_0 - x_- < X < x_0 + x_+]$ . In our experiments, we used  $x_- = x_+ = 5$ ; that is, a range of  $[-5, +5]$  around  $x_0$ , because it gave good results, but there is no absolute and definitive best range. The second step simply consists in evaluating the quantile function of  $P[Y|x_0 - x_- < X < x_0 + x_+]$  at some threshold. The reader can refer to equation 10 for the definition of the quantile function. In our experiments, we used a threshold of 80%, (i.e., we computed  $Q(0.8)$  with the quantile() R function), but the choice of the threshold should be made at the discretion of the user, depending on the desired final interpretation. In plain English, the threshold we selected returns the risk based on worst possible outcomes that is only exceeded in 20% of cases for the particular value of risk

based on real outcomes computed at the first step. Finally, by categorizing this value into the ranges of risk based on worst possible outcomes provided in Table 7, we are able to provide understandable and actionable insight with respect to the most likely risk escalation scenario.

These steps are illustrated for two simple construction situations in Table 8. For comparison, we also show the range of risk based on real outcomes (provided in Table 5) in which  $x_0$  falls.

**Table 8. Illustration of the proposed risk escalation estimation technique**

Step 1: PRIOR EVIDENCE attributes	risk based on real outcomes ( $x_0$ ) and associated range*	Step 2: CONDITIONAL QUANTILE ESTIMATE estimate $Q(0.8)$ of risk based on worst potential outcomes and associated range**
hazardous substance, confined workspace	$590 + 115 = 705$	Extreme
hammer, lumber hand size pieces	$5 + 53 = 58$ 7	Medium Low

\* based on the ranges proposed in Table 5

\*\* based on the ranges proposed in Table 7

## LIMITATIONS

Since the entire process of computing risk values is data driven, the final risk values of the attributes are expected to change from one injury report database to another, and from one set of exposure values to another, even though the distributions of safety risk based on real and worst potential outcomes are expected to remain the same (i.e., heavy-tailed). Also, in this study, we used a rather small dataset (final size of 814 injury reports) to provide a proof of concept for our methodology. With larger datasets, more attributes would play a role in characterizing the different regimes presented in Figure 4, and their respective signature would therefore enjoy a higher resolution.

## **CONCLUSIONS AND RECOMMENDATIONS**

In the first part of this paper, we proposed a methodology to compute univariate and bivariate construction safety risk from attributes and outcomes extracted from raw textual injury reports (i.e., candid observations of the jobsite at injury time). We then showed the empirical probability distribution of construction safety risk to be strikingly similar to that of earthquake, ocean waves, asteroid impact, flood magnitude, and other natural phenomena. Motivated by this finding, we posited that construction safety risk may benefit from being studied in a fully empirical fashion, and introduced data-driven, nonparametric univariate and bivariate modeling and stochastic simulation schemes.

Our approaches were inspired by the state-of-the-art in hydroclimatology and insurance, and are respectively based on Kernel Density Estimators and empirical Copulas. Our nonparametric and empirical data-driven techniques are free of any model fitting, parameter tuning, or assumption making. Therefore, they can be used as a way to ground risk-based safety-related decisions under uncertainty on objective empirical data far exceeding the personal history of even the most experienced safety or project manager. Additionally, the combined use of the attribute-based framework and raw injury reports as the foundation of our approach allows the user to escape the limitations of traditional construction safety risk analysis techniques that are segmented and rely on subjective data. Finally, the attribute-based nature of our methodology enables easy integration with building information modeling (BIM) and work packaging.

We believe this study gives promising evidence that transitioning from an opinion-based and qualitative discipline to an objective, empirically grounded quantitative science would be highly beneficial to construction safety research. Just like the accurate modeling and simulation of natural phenomena such as streamflow, precipitation or wind speed is indispensable to successful structure dimensioning or water reservoir management in Civil engineering, the underlying assumption is that improving construction safety calls for the accurate quantitative modeling, simulation, and assessment of safety risk.

One interesting finding obtained on the data set we used to test our methodology is that central risk shapers are attributes related to high energy levels (e.g., hazardous substance, machinery, forklift) and to human behavior (e.g., improper security of tools, lifting/pulling/manual handling). We remind the reader that the risk values based on real and worst potential outcomes are reported for all attributes in Table 1.

The analyst should decide whether to split the injury report database based on industry branches in which the company is involved, and whether to consider overall exposure values or exposure values per discipline. In any case, interpretations of the risk scores remain valid as long as they are made within the domain from which originated the reports and the exposure values. The former allows to identify differences in risk profiles from one industry discipline to another and to obtain a final product tailored to a particular branch. On the other hand, the latter gives the big picture at the overall company level.

Also, there is currently no automated way to extract real and worst possible severity from a given textual injury report, and it is therefore necessary to have human coders perform the task, which is a costly and lengthy process. Future research should address this issue.

## **ACKNOWLEDGMENTS**

Sincere thanks go to Prof. Arthur Charpentier for his kind help on nonparametric copula density estimation and on the bivariate smoothed bootstrap, and to Prof. Carl Scarrott for the insights provided on dynamic mixture modeling.

## **CHAPTER 6: CONCLUSIONS AND NEXT STEPS**

Traditional construction safety research is limited as it was built on the assumption of independence of tasks and is primarily based upon expert opinion and/or other subjective, aggregated, or secondary data. The refined, more mature version of the attribute-based framework that we used laid the foundations for addressing both limitations, by showing possible the extraction of universal and structured safety information from raw, unstructured injury reports. However, the large scale use of the framework was impossible due to the high costs of manual content analysis.

In the second chapter of this dissertation, we tested the proposition that manual content analysis could be eliminated using NLP. Results show that our system is capable of scanning naturally occurring, unstructured textual injury reports for various fundamental attributes and safety outcomes with high recall (0.97) and precision (0.95) rates. This unlocked the full potential of the attribute-based framework by enabling its widespread application.

In Chapter 3, we illustrated how unsupervised Machine Learning can be used to extract knowledge from the attribute data extracted by our NLP tool. Notably, we represented attribute data sets as weighted undirected graphs and used graph mining to identify key players, cluster precursors into groups, and highlight incompatibilities among attributes. We also applied hierarchical clustering to the same purposes, with promising results.

Chapter 4 focused on the application of two state-of-the art supervised machine learning (ML) algorithms, Random Forest (RF) and Stochastic Gradient Tree Boosting (SGTB), to attribute and outcome data. The resulting models reached high predictive skill, suggesting that construction injuries do not occur in a chaotic fashion, but that rather there are underlying signals to be captured.

Finally, in Chapter 5, we proposed a methodology to compute univariate and bivariate construction safety risk from attributes and outcomes. After noticing that construction safety follows the same frequency-

magnitude distribution as many natural phenomena, we introduced simple yet powerful modeling and stochastic simulation algorithms that can be used by practitioners to better estimate and provision for safety risk.

Overall, our proposed suite of methods, based on binary attributes, is well suited for integration with systems such as BIM, any data-driven technology, and many safety planning activities, including those that take place at the work site. It can be used to support a longitudinal approach of hazard identification and safety management that supports proactive decision-making and provides information with increasing fidelity as project planning matures.

We hope to have shown that construction safety could greatly benefit from being considered an empirically grounded quantitative science

Future research should investigate the use of Machine Learning to complement the hard human-devised rules of the NLP tool with more flexibility, and automatically enrich the various keyword dictionaries. Additionally, a next step on the way to 100% accuracy could consist in automatically detecting the errors made by the tool with data mining methods such as hierarchical clustering. The errors could then be fixed manually.

Also, the theory introduced, which posits that construction accidents are induced by perturbations in underlying networks of fundamental attributes, is promising but needs additional work to be further clarified and delineated. Specifically, attribute networks of injury cases should be compared with that of “non-injury” cases. Such data is also required to generate unconditional injury predictions. The recording of “non-accident” cases could be performed by future research via making random observations of construction situations at times when no injury occurs.

Regarding the predictive models, future research should try to enrich the current list of precursors with additional information, such as the energy level in the environment or worker situational awareness. The former could improve skill for outcomes like injury severity, while the latter could help in making unconditional predictions (occurrence of an accident or not). However, large-scale gathering of this type of information remains a challenge as it does not seem to be easily accessible from text. Especially, situational awareness is hard to assess objectively, and if possible at all, systematic assessments would need to be conducted onsite and for many employees in an almost real-time fashion, which is a major barrier.

## REFERENCES

Ahn, S., Kim, J. H., & Ramaswami, V. (2012). A new class of models for heavy tailed distributions in finance and insurance risk. *Insurance: Mathematics and Economics*, 51(1), 43-52.

Akbani, R., Kwek, S., & Japkowicz, N. (2004). Applying support vector machines to imbalanced datasets. In *Machine Learning: ECML 2004* (pp. 39-50). Springer Berlin Heidelberg.

Al Qady, M., & Kandil, A. (2014). Automatic clustering of construction project documents based on textual similarity. *Automation in Construction*, 42, 36-49.

Albert, A., Hallowell, M. R., Kleiner, B., Chen, A., & Golparvar-Fard, M. (2014). Enhancing construction hazard recognition with high-fidelity augmented virtuality. *Journal of Construction Engineering and Management*, 140(7).

Alexander, D., Hallowell, M., & Gambatese, J. (2015). Energy-based safety risk management: using hazard energy to predict injury severity. In *Proceedings of the 5th International/11th Construction Specialty Conference*, Vancouver.

Almeida, J. A. S., Barbosa, L. M. S., Pais, A. A. C. C., & Formosinho, S. J. (2007). Improving hierarchical cluster analysis: A new method with outlier detection and automatic clustering. *Chemometrics and Intelligent Laboratory Systems*, 87(2), 208-217

Almén, L., Larsson, T. J., & Thunqvist, E. L. (2012). The Influence of the Designer on the Risk of Falling from Heights and of Exposure to Excessive Workloads on two Construction Sites. *Safety Science Monitor*, 16(1), 2-7.

Alsamadani, R., Hallowell, M., & Javernick-Will, A. N. (2013). Measuring and modelling safety communication in small work crews in the US using social network analysis. *Construction Management and Economics*, 31(6), 568-579.

Arciszewski, Tomasz and Mumtaz, Usman (1993). Applications of Machine Learning to Construction Safety. In *Proceedings of the International Conference on Management of Information Technology for Construction, Singapore*.

Arciszewski, Tomasz, Mumtaz Usman, and John Gleichman (1991). Construction accident analysis: the inductive learning approach. *Preparing for Construction in the 21<sup>st</sup> Century* 253-258.

Bai, X. (2011). Predicting consumer sentiments from online text. *Decision Support Systems*, 50(4), 732-742.

Baradan, S., & Usman, M. A. (2006). Comparative injury and fatality risk analysis of building trades. *Journal of construction engineering and management*, 132(5), 533-539.

Barbella, D., Benzaid, S., Christensen, J. M., Jackson, B., Qin, X. V., & Musicant, D. R. (2009, July). Understanding Support Vector Machine Classifications via a Recommender System-Like Approach. In *DMIN* (pp. 305-311).

Bárdossy, A., & Pegram, G. (2014). Infilling missing precipitation records—A comparison of a new copula-based method with other techniques. *Journal of hydrology*, 519, 1162-1170.

Beleites, C., Neugebauer, U., Bocklitz, T., Krafft, C., & Popp, J. (2013). Sample size planning for classification models. *Analytica chimica acta*, 760, 25-33.

Benzécri, J. P. (1973). L'analyse des données. Tome 1: La taxinomie. Tome 2: L'analyse des correspondances.

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *The Journal of Machine Learning Research*, 13(1), 281-305.

Biau, G. (2012). Analysis of a random forests model. *The Journal of Machine Learning Research*, 13(1), 1063-1095.

Biau, G., Devroye, L., & Lugosi, G. (2008). Consistency of random forests and other averaging classifiers. *The Journal of Machine Learning Research*, 9, 2015-2033.

Blondel, V. D., Guillaume, J. L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10), P10008.

Bonacich, P. (1972). Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology*, 2(1), 113-120.

Booth, A., Gerding, E., & McGroarty, F. (2014). Automated trading with performance weighted random forests and seasonality. *Expert Systems with Applications*, 41(8), 3651-3661.

Borgatti, S. P. (2005). Centrality and network flow. *Social networks*, 27(1), 55-71.

Breiman, L. (1996a). Bagging predictors. *Machine learning*, 24(2), 123-140.

Breiman, L. (1996b). *Out-of-bag estimation* (pp. 1-13). Technical report, Statistics Department, University of California Berkeley, Berkeley CA 94708, 1996b. 33, 34.

Breiman, L. (1998). Arcing classifier (with discussion and a rejoinder by the author). *The annals of statistics*, 26(3), 801-849.

Breiman, L. (2001a). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical Science*, 16(3), 199-231.

Breiman, L. (2001b). Random forests. *Machine learning*, 45(1), 5-32.

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

Brown, G. (2010). Ensemble learning. In *Encyclopedia of Machine Learning* (pp. 312-320). Springer US.

Buckland, M. K., & Gey, F. C. (1994). The relationship between recall and precision. *JASIS*, 45(1), 12-19.

Bullmore, E., & Sporns, O. (2009). Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience*, 10(3), 186-198.

Bureau of Labor Statistics (2015). Occupational Injuries/Illnesses and Fatal Injuries Profiles.  
<http://www.bls.gov/news.release/pdf/cfoi.pdf>.

Bureau of Labor Statistics (BLS) (2013). "Census of Fatal Occupational Injuries (CFOI) - Current and Revised Data." Accessed August 21, 2015. <http://www.bls.gov/iif/oshcfoi1.htm>.

Caldas, C. H., & Soibelman, L. (2003). Automating hierarchical document classification for construction management information systems. *Automation in Construction*, 12(4), 395-406.

Capen, E. C. (1976). The Difficulty of Assessing Uncertainty (includes associated papers 6422 and 6423 and 6424 and 6425). *Journal of Petroleum Technology*, 28(08), 843-850.

Carter, G., & Smith, S. D. (2006). Safety hazard identification on construction projects. *Journal of Construction Engineering and Management*, 132(2), 197-205.

Caruana, R., & Niculescu-Mizil, A. (2006, June). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning* (pp. 161-168). ACM.

Chae, Soungho, and Tomohiro Yoshida. (2008). A Study of Safety Management Using Working Area Information on Construction Site. In *Proceedings of the 25th International Symposium on Automation and Robotics in Construction*, June, 292–99.

Charpentier, A. (2006). Structures de dépendance et résultats limites avec applications en finance assurance (Doctoral dissertation, ENSAE ParisTech).

Charpentier, A., & Oulidi, A. (2010). Beta kernel quantile estimators of heavy-tailed loss distributions. *Statistics and computing*, 20(1), 35-55.

Charpentier, A., Fermanian, J. D., & Scaillet, O. (2007). The estimation of copulas: Theory and practice. *Copulas: from theory to applications in finance*, 35-62.

Chawla, N. V. (2005). Data mining for imbalanced datasets: An overview. In *Data mining and knowledge discovery handbook* (pp. 853-867). Springer US.

Chen, C., Liaw, A., & Breiman, L. (2004). Using random forest to learn imbalanced data. *University of California, Berkeley*.

Cheng, M. Y., & Wu, Y. W. (2009). Evolutionary support vector machine inference system for construction management. *Automation in Construction*, 18(5), 597-604.

Cheng, M. Y., Peng, H. S., Wu, Y. W., & Chen, T. L. (2010). Estimate at completion for construction projects using evolutionary support vector machine inference model. *Automation in Construction*, 19(5), 619-629.

Cheng, M. Y., Peng, H. S., Wu, Y. W., & Liao, Y. H. (2011). Decision making for contractor insurance deductible using the evolutionary support vector machines inference model. *Expert Systems with Applications*, 38(6), 6547-6555.

Cherubini, U., Luciano, E., & Vecchiato, W. (2004). *Copula methods in finance*. John Wiley & Sons.

Chinowsky, P. S., Diekmann, J., & O'Brien, J. (2009). Project organizations as social networks. *Journal of Construction Engineering and Management*, 136, SPECIAL ISSUE: Governance and Leadership Challenges of Global Construction, 452–458.

Chowdhury, G. G. (2003). Natural language processing. *Annual review of information science and technology*, 37(1), 51-89.

Ciribini, A. L. C., A. Gottfried, M. L. Trani, and L. Bergamini. (2011). 4D Modelling and Construction Health and Safety Planning. In *Proceedings of the 6th International Structural Engineering and Construction Conference - Modern Methods and Advances in Structural Engineering and Construction*, 467–71. Zurich, Switzerland.

Claesen, M., & De Moor, B. (2015). Hyperparameter Search in Machine Learning. *arXiv preprint arXiv:1502.02127*.

Clauset, A., Newman, M. E., & Moore, C. (2004). Finding community structure in very large networks. *Physical review E*, 70(6), 066111.

Coles, S., Bawa, J., Trenner, L., & Dorazio, P. (2001). *An introduction to statistical modeling of extreme values* (Vol. 208). London: Springer.

Collins, F. S., Green, E. D., Guttmacher, A. E., & Guyer, M. S. (2003). A vision for the future of genomics research. *Nature*, 422(6934), 835-847.

Collins, R., Zhang, S., Kim, K., & Teizer, J. (2014). Integration of safety risk factors in BIM for scaffolding construction. *Proc. ICCCBE*.

Costin, Aaron, Nipesh Pradhananga, and Jochen Teizer. (2014). Passive RFID and BIM for Real-Time Visualization and Location Tracking. In *Construction Research Congress 2014*, 169–78. American Society of Civil Engineers.

CPWR 2013 – The Center for Construction Research and Training, produced with support from the National Institute for Occupational Safety and Health grant number OH009762.

Crovella, Mark E., and Azer Bestavros. Explaining World Wide Web Traffic Self-Similarity. Boston University Computer Science Department, 1995. <http://dcommon.bu.edu/xmlui/handle/2144/1574>.

Csardi, G., & Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal, Complex Systems*, 1695(5), 1-9.

del Río, S., López, V., Benítez, J. M., & Herrera, F. (2014). On the use of MapReduce for imbalanced big data using Random Forest. *Information Sciences*, 285, 112-137.

del Sol, A., Balling, R., Hood, L., & Galas, D. (2010). Diseases as network perturbations. *Current Opinion in Biotechnology*, 21(4), 566-571.

Desvignes, M. (2014). *Requisite empirical risk data for integration of safety with advanced technologies and intelligent systems* (Master thesis, University of Colorado at Boulder).

Díaz-Uriarte, R., & de Andrés, S. A. (2005). Variable selection from random forests: application to gene expression data. *arXiv preprint q-bio/0503025*.

Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM*, 55(10), 78-87.

Domino, K., Błachowicz, T., & Ciupak, M. (2014). The use of copula functions for predictive analysis of correlations between extreme storm tides. *Physica A: Statistical Mechanics and its Applications*, 413, 489-497.

Dorogovtsev, S. N., & Mendes, J. F. (2013). Evolution of networks: From biological nets to the Internet and WWW. *Oxford University Press*.

Elith, J., Leathwick, J. R., & Hastie, T. (2008). A working guide to boosted regression trees. *Journal of Animal Ecology*, 77(4), 802-813.

Esmaeili, B. (2012). *Identifying and quantifying construction safety risks at the attribute level* (Doctoral dissertation, University of Colorado at Boulder).

Esmaeili, B., & Hallowell, M. (2012, May). Attribute-based risk model for measuring safety risk of struck-by accidents. In *Construction Research Congress* (pp. 289-298).

Esmaeili, B., & Hallowell, M. R. (2011a). Diffusion of safety innovations in the construction industry. *Journal of Construction Engineering and Management*, 138(8), 955-963.

Esmaeili, B., & Hallowell, M. R. (2011b). Using network analysis to model fall hazards on construction projects. *Safety and Health in Construction, CIB W*, 99, 24-26.

Esmaeili, B., Hallowell, M. R., & Rajagopalan, B. (2015a). Attribute-Based Safety Risk Assessment. I: Analysis at the Fundamental Level. *Journal of Construction Engineering and Management*, 04015021.

Esmaeili, B., Hallowell, M. R., & Rajagopalan, B. (2015b). Attribute-Based Safety Risk Assessment. II: Predicting Safety Outcomes Using Generalized Linear Models. *Journal of Construction Engineering and Management*, 04015022.

Everett, J. (1999). Overexertion Injuries in Construction. *Journal of Construction Engineering and Management*, 125 (2): 109–14.

Fang, C., Marle, F., Zio, E., & Bocquet, J. C. (2012). Network theory-based analysis of risk interactions in large engineering projects. *Reliability Engineering & System Safety*, 106, 1-10.

Firth, J. R. (1957). *Papers in linguistics, 1934-1951*. Oxford University Press.

Fleming, M. A. (2009). Hazard recognition. *By Design*, ASSE, 11-15.

Fortunato, S. (2010). Community detection in graphs. *Physics Reports*, 486(3), 75-174.

Francis, L., & Flynn, M. (2010). Text mining handbook. In *Casualty Actuarial Society E-Forum, Spring 2010* (Vol. 1).

Franz, K. J., & Sorooshian, S. (2002). Verification of National Weather Service Probabilistic Hydrologic Forecasts. *University of Arizona, report prepared for the National Weather Service*.

Freeman, L. C. (1979). Centrality in social networks conceptual clarification. *Social networks*, 1(3), 215-239.

Freiman, M. H. (2010). Using random forests and simulated annealing to predict probabilities of election to the baseball hall of fame. *Journal of Quantitative Analysis in Sports*, 6(2).

Freund, Y., & Schapire, R. E. (1995, January). A decision-theoretic generalization of on-line learning and an application to boosting. In *Computational learning theory* (pp. 23-37). Springer Berlin Heidelberg.

Freund, Y., & Schapire, R. E. (1996, July). Experiments with a new boosting algorithm. In *ICML* (Vol. 96, pp. 148-156).

Freund, Y., Schapire, R., & Abe, N. (1999). A short introduction to boosting. *Journal of Japanese Society For Artificial Intelligence*, 14(771-780), 1612.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189-1232.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4), 367-378.

Friedman, J., Hastie, T., & Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 28(2), 337-407.

Frigessi, A., Haug, O., & Rue, H. (2002). A dynamic mixture model for unsupervised tail estimation without threshold selection. *Extremes*, 5(3), 219-235.

Fruchterman, Thomas M. J.; Reingold, Edward M. (1991), Graph Drawing by Force-Directed Placement, *Software – Practice & Experience* (Wiley) 21 (11): 1129–1164, doi:10.1002/spe.4380211102.

Fullerton, Clare E., Ben S. Allread, and Jochen Teizer. (2009). Pro-Active-Real-Time Personnel Warning System. In *Construction Research Congress 2009*, 31–40. American Society of Civil Engineers.

Furrer, E. M., & Katz, R. W. (2008). Improving the simulation of extreme precipitation events by stochastic weather generators. *Water Resources Research*, 44(12).

Gambatese, J. A. (2008). Research issues in prevention through design. *Journal of safety research*, 39(2), 153-156.

Genuer, R., Poggi, J. M., & Tuleau-Malot, C. (2010). Variable selection using random forests. *Pattern Recognition Letters*, 31(14), 2225-2236.

Gilleland, E., & Katz, R. W. (2011). New software to analyze how extremes change over time. *Eos, Transactions American Geophysical Union*, 92(2), 13-14.

Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1-12.

Goddard, L., Barnston, A. G., & Mason, S. J. (2003). Evaluation of the IRI's "Net assessment" seasonal climate forecasts: 1997-2001. *Bulletin of the American Meteorological Society*, 84(12), 1761-1781.

Goedert, J., and P. Meadati. (2008). Integrating Construction Process Documentation into Building Information Modeling. *Journal of Construction Engineering and Management*, 134 (7): 509–16.

Goutte, C., & Gaussier, E. (2005). A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In *Advances in information retrieval* (pp. 345-359). Springer Berlin Heidelberg.

Greenacre, M. (2007). Correspondence analysis in practice. *CRC press*.

Greg Ridgeway with contributions from others (2015). gbm: Generalized Boosted Regression Models. R package version 2.1.1. <http://CRAN.R-project.org/package=gbm>

Grimmer, J., & Stewart, B. M. (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis*, mps028.

Gustafson, Per E. (1998). Gender Differences in Risk Perception: Theoretical and Methodological Perspectives. *Risk Analysis*, 18 (6): 805–11.

Haddon, W. (1973). Energy damage and the ten countermeasure strategies. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 15(4), 355-366.

Halkidi, M., Batistakis, Y., & Vazirgiannis, M. (2001). On clustering validation techniques. *Journal of Intelligent Information Systems*, 17(2), 107-145.

Hallowell, M. R. (2008). *A formal model for construction safety and health risk management*. (Doctoral dissertation, Oregon State University)

Hallowell, M. R., & Gambatese, J. A. (2009a). Activity-based safety risk quantification for concrete formwork construction. *Journal of Construction Engineering and Management*, 135(10), 990-998.

Hallowell, M. R., & Gambatese, J. A. (2009b). Qualitative research: Application of the Delphi method to CEM research. *Journal of construction engineering and management*, 136(1), 99-107.

Hallowell, M., Esmaeli, B., & Chinowsky, P. (2011). Safety risk interactions among highway construction work tasks. *Construction Management and Economics*, 29(4), 417-429.

Hammad, A, Cheng Zhang, S. Setayeshgar, and Y. Asen. (2012). Automatic Generation of Dynamic Virtual Fences as Part of BIM-Based Prevention Program for Construction Safety. In *Simulation Conference (WSC)*, Proceedings of the 2012 Winter, 1–10.

Hastie, T., Tibshirani, R., Friedman, J. (2009). The elements of statistical learning (Vol. 2, No. 1). *New York: springer.*

He, H., & Garcia, E. (2009). Learning from imbalanced data., *IEEE Transactions on Knowledge and Data Engineering*, 21(9), 1263-1284.

Helander, M. G. (1991). Safety hazards and motivation for safe work in the construction industry. *International Journal of Industrial Ergonomics*, 8(3), 205-223.

Hindle, D. (1989, June). Acquiring disambiguation rules from text. In *Proceedings of the 27th annual meeting on Association for Computational Linguistics* (pp. 118-125).

Hinze, J. W., & Teizer, J. (2011). Visibility-related fatalities related to construction equipment. *Safety science*, 49(5), 709-718.

Hinze, J., Devenport, J. N., & Giang, G. (2006). Analysis of construction worker injuries that do not result in lost time. *Journal of Construction engineering and management*, 132(3), 321-326.

Hopkins, D. J., & King, G. (2010). A method of automated nonparametric content analysis for social science. *American Journal of Political Science*, 54(1), 229-247.

Hsu, J. Y. (2013). Content-based text mining technique for retrieval of CAD documents. *Automation in Construction*, 31, 65-74.

Hu, Y., & Scarrott, C. J. (2013). evmix: An R package for extreme value mixture modelling, threshold estimation and boundary corrected kernel density estimation.

Huang, S., Ernberg, I., & Kauffman, S. (2009, September). Cancer attractors: a systems view of tumors from a gene network dynamics and developmental perspective. In *Seminars in cell & developmental biology* (Vol. 20, No. 7, pp. 869-876). Academic Press.

Huang, X., and J. Hinze (2003). Analysis of Construction Worker Fall Accidents. *Journal of Construction Engineering and Management* 129 (3): 262-71.

Hull, J. (2006). Defining copulas. *Risk*, 19(10), 62-64.

Husson, F., Josse, J., & Pages, J. (2010). Principal component methods-hierarchical clustering-partitional clustering: why would we need to choose for visualizing data. *Applied Mathematics Department*.

i Cancho, R. F., & Solé, R. V. (2001). The small world of human language. *Proceedings of the Royal Society of London B: Biological Sciences*, 268(1482), 2261-2265.

Iacobucci, Dawn (ed.) (2001), Journal of Consumer Psychology's Special issue on Methodological and Statistical Concerns of the Experimental Behavioral Researcher, 10 (1&2), Mahwah, NJ: Lawrence Erlbaum Associates, 71-73

Ingo Feinerer, Kurt Hornik, and David Meyer (2008). "Text Mining Infrastructure in R." *Journal of Statistical Software* 25(5): 1-54. URL: <http://www.jstatsoft.org/v25/i05/>.

Jagger, T.H., J.B. Elsner, and M.A. Saunders, 2008: "Forecasting U.S. insured hurricane losses." In Climate Extremes and Society, edited by H.F. Diaz and R.J. Murnane, Cambridge University Press, pp. 189-208

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (p. 6). *New York: Springer.*

Jannadi, O., and S. Almishari (2003). Risk Assessment in Construction. *Journal of Construction Engineering and Management* 129 (5): 492–500.

Joachims, T. (1998). *Text categorization with support vector machines: Learning with many relevant features* (pp. 137-142). Springer Berlin Heidelberg.

Jolliffe, I. (1986). Principal Components Analysis, *Springer, Verlag.*

Jones, M. C. (1993). Simple boundary correction for kernel density estimation. *Statistics and Computing*, 3(3), 135-146.

Jonsson, P. F., Cavanna, T., Zicha, D., & Bates, P. A. (2006). Cluster analysis of networks generated through homology: automatic identification of important protein communities involved in cancer metastasis. *BMC bioinformatics*, 7(1), 2.

Kahneman, D., & Tversky, A. (1982). On the study of statistical intuitions. *Cognition*, 11(2), 123-141.

Kaner, I., Sacks, R., Kassian, W., & Quitt, T. (2008). Case studies of BIM adoption for precast concrete design by mid-sized structural engineering firms. ITcon Vol. 13, Special Issue Case studies of BIM use, pg. 303-323, <http://www.itcon.org/2008/21>

Karatzoglou, A., & Feinerer, I. (2010). Kernel-based machine learning for fast text mining in R. *Computational Statistics & Data Analysis*, 54(2), 290-297.

Karrer, B., Levina, E., & Newman, M. E. (2007). Robustness of community structure in networks. *arXiv preprint arXiv:0709.2108*.

Katz, Richard W., Marc B. Parlange, and Philippe Naveau. "Statistics of extremes in hydrology." *Advances in water resources* 25, no. 8 (2002): 1287-1304.

Kauffman, S. A. (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of theoretical biology*, 22(3), 437-467.

Khalilia, M., Chakraborty, S., & Popescu, M. (2011). Predicting disease risks from highly imbalanced data using random forest. *BMC medical informatics and decision making*, 11(1), 51.

Kohavi, R. (1995, August). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai* (Vol. 14, No. 2, pp. 1137-1145).

Kuechler, W. L. (2007). Business applications of unstructured text. *Communications of the ACM*, 50(10), 86-93.

Lall, U., Rajagopalan, B., & Tarboton, D. G. (1996). A nonparametric wet/dry spell model for resampling daily precipitation. *Water Resources Research*, 32(9), 2803-2823.

Lam, K. C., Palaneeswaran, E., & Yu, C. Y. (2009). A support vector machine model for contractor prequalification. *Automation in Construction*, 18(3), 321-329.

Lancichinetti, A., Radicchi, F., Ramasco, J. J., & Fortunato, S. (2011). Finding statistically significant communities in networks. *PloS one*, 6(4), e18961.

Lebedev, A. V., Westman, E., Van Westen, G. J. P., Kramberger, M. G., Lundervold, A., Aarsland, D., ... & AddNeuroMed consortium. (2014). Random Forest ensembles for detection and prediction of Alzheimer's disease with a good between-cohort robustness. *NeuroImage: Clinical*, 6, 115-125.

Liang, T. H., & Lin, J. B. (2014). A two-stage segment and prediction model for mortgage prepayment prediction and management. *International Journal of Forecasting*, 30(2), 328-343.

Liaw and M. Wiener (2002). Classification and Regression by randomForest. *R News* 2(3), 18—22

Lichman, M. (2013). UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science.

Liddy, E. D. (2001). Natural language processing.

Lin, T. H., Liu, C. H., Tsai, M. H., & Kang, S. C. (2014). Using Augmented Reality in a Multiscreen Environment for Construction Discussion. *Journal of Computing in Civil Engineering*, 04014088.

Lingard, H. (2013). Occupational health and safety in the construction industry. *Construction Management and Economics*, 31(6), 505-514.

Lombard, M., Snyder-Duch, J., & Bracken, C. C. (2002). Content analysis in mass communication: Assessment and reporting of intercoder reliability. *Human communication research*, 28(4), 587-604.

Louppe, G. (2014). Understanding Random Forests: From Theory to Practice. *arXiv preprint arXiv:1407.7502*.

Loureiro, A., Torgo, L., & Soares, C. (2004). Outlier detection using clustering methods: a data cleaning application. In *Proceedings of KDNet Symposium on Knowledge-based systems for the Public Sector*.

Malamud, Bruce D. "Tails of natural hazards." Physics World 17, no. 8 (2004): 31-35.

Malamud, Bruce D., and Donald L. Turcotte. "The applicability of power-law frequency statistics to floods." Journal of Hydrology 322, no. 1 (2006): 168-180.

Manning, C. D. (1999). *Foundations of statistical natural language processing*. H. Schütze (Ed.). MIT press.

Menéndez, M., Méndez, F. J., Losada, I. J., & Graham, N. E. (2008). Variability of extreme wave heights in the northeast Pacific Ocean based on buoy measurements. *Geophysical Research Letters*, 35(22).

Miller, George A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review*, 63(2), 81-97.

Moon, Y. I., Rajagopalan, B., & Lall, U. (1995). Estimation of mutual information using kernel density estimators. *Physical Review E*, 52(3), 2318-2321.

Moselhi, O., Hegazy, T., & Fazio, P. (1991). Neural networks as tools in construction. *Journal of Construction Engineering and Management*.

Murthy, K. V. S. (1995). *On growing better decision trees from data*. (Doctoral dissertation, Johns Hopkins University).

Navon, R., & Kolton, O. (2006). Model for automated monitoring of fall hazards in building construction. *Journal of Construction Engineering and Management*, 132(7), 733-740.

NCAR - Research Applications Laboratory (2015). verification: Weather Forecast Verification Utilities. R package version 1.42. <http://CRAN.R-project.org/package=verification>

Newman, M. E. (2006). Finding community structure in networks using the eigenvectors of matrices. *Physical review E*, 74(3), 036104.

Occupational Injury and Illness Classification Manual Version 2.0, U.S. Department of Labor, Bureau of Labor Statistics, September 2010 ([http://www.bls.gov/iif/oiics\\_manual\\_2010.pdf](http://www.bls.gov/iif/oiics_manual_2010.pdf))

Opitz, D., & Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 169-198.

Palamara, F., Piglione, F., & Piccinini, N. (2011). Self-Organizing Map and clustering algorithms for the analysis of occupational accident databases. *Safety science*, 49(8), 1215-1230.

Papalexiou, S. M., D. Koutsoyiannis, and C. Makropoulos. "How extreme is extreme? An assessment of daily rainfall distribution tails." *Hydrology and Earth System Sciences* 17, no. 2 (2013): 851-862.

Pareto V. *Cours d'économie politique*. Geneva, Switzerland: Droz; 1896.

Pereira, F. C., Rodrigues, F., & Ben-Akiva, M. (2013). Text analysis in incident duration prediction. *Transportation Research Part C: Emerging Technologies*, 37, 177-192.

Pinto, Carla, A. Mendes Lopes, and J. A. Machado. "A review of power laws in real life phenomena." *Communications in Nonlinear Science and Numerical Simulation* 17, no. 9 (2012): 3558-3578.

Pons, P., & Latapy, M. (2005). Computing communities in large networks using random walks. In *Computer and Information Sciences-ISCIS 2005* (pp. 284-293). Springer Berlin Heidelberg.

Ponticelli, S., O'Brien, W. J., & Leite, F. (2015). Advanced work packaging as emerging planning approach to improve project performance: case studies from the industrial construction sector.

Prabowo, R., & Thelwall, M. (2009). Sentiment analysis: A combined approach. *Journal of Informetrics*, 3(2), 143-157.

Prades Villanova, M. (2014). *Attribute-based Risk Model for Assessing Risk to Industrial Construction Tasks* (Master thesis, University of Colorado at Boulder).

R Core Team (2014). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>.

Radvansky, G. A., Krawietz, S. A., & Tamplin, A. K. (2011). Walking through doorways causes forgetting: Further explorations. *The Quarterly Journal of Experimental Psychology*, 64(8), 1632-1645.

Raileanu, L. E., & Stoffel, K. (2004). Theoretical comparison between the gini index and information gain criteria. *Annals of Mathematics and Artificial Intelligence*, 41(1), 77-93.

Rajagopalan, B., Grantz, K., Regonda, S., Clark, M., & Zagona, E. (2005). Ensemble streamflow forecasting: Methods and applications. *Advances in Water Science Methodologies*, 97-116.

Rajagopalan, B., Lall, U., & Tarboton, D. G. (1997a). Evaluation of kernel density estimation methods for daily precipitation resampling. *Stochastic Hydrology and Hydraulics*, 11(6), 523-547.

Rajagopalan, B., Lall, U., Tarboton, D. G., & Bowles, D. S. (1997b). Multivariate nonparametric resampling scheme for generation of daily weather variables. *Stochastic Hydrology and Hydraulics*, 11(1), 65-93.

Rebollo, J. J., & Balakrishnan, H. (2014). Characterization and prediction of air traffic delays. *Transportation Research Part C: Emerging Technologies*, 44, 231-241.

Reed, William J. "The Pareto, Zipf and other power laws." *Economics Letters* 74, no. 1 (2001): 15-19.

Reichardt, J., & Bornholdt, S. (2006). Statistical mechanics of community detection. *Physical Review E*, 74(1), 016110.

Revolution Analytics and Steve Weston (2014). “doParallel: Foreach parallel adaptor for the parallel package.” R package version 1.0.8. <http://CRAN.R-project.org/package=doParallel>

Ridgeway, G. (2007). Generalized Boosted Models: A guide to the gbm package. *Update*, 1(1).

Rivas, T., Paz, M., Martín, J. E., Matías, J. M., García, J. F., & Taboada, J. (2011). Explaining and predicting workplace accidents using data-mining techniques. *Reliability Engineering & System Safety*, 96(7), 739-747.

Robnik-Šikonja, M. (2004). Improving random forests. In *Machine Learning: ECML 2004* (pp. 359-370). Springer Berlin Heidelberg.

Rose, P. R. (1987). Dealing with risk and uncertainty in exploration: how can we improve? *AAPG Bulletin*, 71(1), 1-16.

Rousseau, F., Kiagias, E., & Vazirgiannis, M. (2015). Text Categorization as a Graph Classification Problem. In *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing* (Vol. 1, pp. 1702-1712).

Ruhnau, B. (2000). Eigenvector-centrality—a node-centrality? *Social networks*, 22(4), 357-365.

Sacks, R., Rozenfeld, O., & Rosenfeld, Y. (2009). Spatial and temporal exposure to safety hazards in construction. *Journal of construction engineering and management*, 135(8), 726-736.

Sagae, K., & Lavie, A. (2003). Combining Rule-based and Data-driven Techniques for Grammatical Relation Extraction in Spoken Language. In *Proceedings of the Eighth International Workshop in Parsing Technologies. Nancy, France.*

Salvadori, G., De Michele, C., Kottekoda, N. T., & Rosso, R. (2007). Extremes in nature: an approach using copulas (Vol. 56). Springer Science & Business Media.

Sandri, M., & Zuccolotto, P. (2006). Variable selection using random forests. In *Data analysis, classification and the forward search* (pp. 263-270). Springer Berlin Heidelberg.

Saporta, G. (2011). Probabilités, analyse des données et statistique. Editions Technip.

Scarrott, C., & MacDonald, A. (2012). A Review of Extreme Value Threshold Estimation and Uncertainty Quantification. *REVSTAT–Statistical Journal*, 10(1), 33-60.

Schaeffer, S. E. (2007). Graph clustering. *Computer Science Review*, 1(1), 27-64.

Seera, M., & Lim, C. P. (2014). A hybrid intelligent system for medical data classification. *Expert Systems with Applications*, 41(5), 2239-2249.

Shapira, A., and B. Lyachin. 2009. "Identification and Analysis of Factors Affecting Safety on Construction Sites with Tower Cranes." *Journal of Construction Engineering and Management* 135 (1): 24–33.

Sharma, A., Tarboton, D. G., & Lall, U. (1997). Streamflow simulation: A nonparametric approach. *Water Resources Research*, 33(2), 291-308.

Shlens, J. (2014). A tutorial on principal component analysis. *arXiv preprint arXiv:1404.1100*.

Silverman, B. W. (1986). Density estimation for statistics and data analysis (Vol. 26). CRC press.

Skibniewski, M., Arciszewski, T., & Lueprasert, K. (1997). Constructability analysis: machine learning approach. *Journal of computing in civil engineering*, 11(1), 8-16.

Sklar, A. (1959), "Fonctions de répartition à n dimensions et leurs marges", Publ. Inst. Statist. Univ. Paris 8: 229–231

Smith, L. I. (2002). A tutorial on principal components analysis. *Cornell University*, USA, 51, 52.

Soibelman, L., & Kim, H. (2002). Data preparation process for construction knowledge generation through knowledge discovery in databases. *Journal of Computing in Civil Engineering*, 16(1), 39–48.

Soibelman, L., Wu, J., Caldas, C., Brilakis, I., & Lin, K. Y. (2008). Management and analysis of unstructured construction data types. *Advanced Engineering Informatics*, 22(1), 15-27.

Son, H., Kim, C., & Kim, C. (2011). Automated color model-based concrete detection in construction-site images by using machine learning algorithms. *Journal of Computing in Civil Engineering*, 26(3), 421-433.

Stelling, J., Sauer, U., Szallasi, Z., Doyle, F. J., & Doyle, J. (2004). Robustness of cellular functions. *Cell*, 118(6), 675-685.

Strobl, C., Malley, J., & Tutz, G. (2009). An introduction to recursive partitioning: rationale, application, and characteristics of classification and regression trees, bagging, and random forests. *Psychological methods*, 14(4), 323.

Sun, Y., Kamel, M. S., Wong, A. K., & Wang, Y. (2007). Cost-sensitive boosting for classification of imbalanced data. *Pattern Recognition*, 40(12), 3358-3378.

Sutton, C. D. (2005). Classification and regression trees, bagging, and boosting. *Handbook of statistics*, 24, 303-329.

Tang, Y., Zhang, Y. Q., Chawla, N. V., & Krasser, S. (2009). SVMs modeling for highly imbalanced classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39(1), 281-288.

Taylor, I. W., Linding, R., Warde-Farley, D., Liu, Y., Pesquita, C., Faria, D., ... & Wrana, J. L. (2009). Dynamic modularity in protein interaction networks predicts breast cancer outcome. *Nature biotechnology*, 27(2), 199-204.

Timofeev, R. (2004). *Classification and regression trees (CART) theory and applications*. (Master thesis, Humboldt University, Berlin).

Tixier, A. J. P., Hallowell, M. R., Albert, A., van Boven, L., & Kleiner, B. M. (2014). Psychological antecedents of risk-taking behavior in construction. *Journal of Construction Engineering and Management*.

Tixer, A. J. P., Hallowell, M. R., Rajagopalan, B., & Bowman, D. (2016a). Automated content analysis for construction safety: A natural language processing system to extract precursors and outcomes from unstructured injury reports. *Automation in Construction*, 62, 45-56.

Tixer, A. J. P., Hallowell, M. R., Rajagopalan, B., & Bowman, D. (2016b). Application of machine learning to construction injury prediction. *Automation in construction*, 69, 102-114.

Towler, E., Rajagopalan, B., Summers, R. S., & Yates, D. (2010). An approach for probabilistic forecasting of seasonal turbidity threshold exceedance. *Water Resources Research*, 46(6).

Tsanas, A., & Xifara, A. (2012). Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy and Buildings*, 49, 560-567.

Tversky, A., & Kahneman, D. (1981). The framing of decisions and the psychology of choice. *Science*, 211(4481), 453-458.

Verma, S., Vieweg, S., Corvey, W. J., Palen, L., Martin, J. H., Palmer, M., ... & Anderson, K. M. (2011, July). Natural Language Processing to the Rescue? Extracting "Situational Awareness" Tweets During Mass Emergency. In *ICWSM*.

Vrac, M., & Naveau, P. (2007). Stochastic downscaling of precipitation: From dry events to heavy rainfalls. *Water resources research*, 43(7).

Wang, Y. Y., Acero, A., Chelba, C., Frey, B. J., & Wong, L. (2002, September). Combination of statistical and rule-based approaches for spoken language understanding. In *INTERSPEECH*.

Ward Jr, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*, 58(301), 236-244.

Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of ‘small-world’ networks. *Nature*, 393(6684), 440-442.

Wei, W. W. S. (1994). *Time series analysis*. Addison-Wesley publ.

Weigel, A. P., Liniger, M. A., & Appenzeller, C. (2007). The discrete Brier and ranked probability skill scores. *Monthly Weather Review*, 135(1), 118-124.

Weiss, G. M. (2004). Mining with rarity: a unifying framework. *ACM SIGKDD Explorations Newsletter*, 6(1), 7-19.

Wilks, D. S. (1995), Statistical Methods in the Atmospheric Sciences, Elsevier, New York.

Wolpert, D. H., & Macready, W. G. (1999). An efficient method to estimate bagging's generalization error. *Machine Learning*, 35(1), 41-55.

Yang, J., Arif, O., Vela, P. A., Teizer, J., & Shi, Z. (2010). Tracking multiple workers on construction sites using video cameras. *Advanced Engineering Informatics*, 24(4), 428-434.

Yang, R. J., & Zou, P. X. (2014). Stakeholder-associated risks and their interactions in complex green building projects: A social network model. *Building and Environment*, 73, 208-222.

Yeh, K., M. Tsai, and S. Kang. (2012). On-Site Building Information Retrieval by Using Projection-Based Augmented Reality. *Journal of Computing in Civil Engineering* 26 (3): 342–55.

Yeung, C. L., Cheung, C. F., Wang, W. M., & Tsui, E. (2014). A knowledge extraction and representation system for narrative analysis in the construction industry. *Expert Systems with Applications*, 41(13), 5710-5722.

Zienkiewics, O. C. (1971). The finite element method in engineering science. *McGraw-Hill, Londres*