Too Little for Big Data?

Jane Huffman Hayes  
Computer Science Department  
University of Kentucky  
Lexington, Kentucky, USA  
hayes@cs.uky.edu

Giulio Antoniol  
École Polytechnique de Montréal  
Montreal, Quebec, Canada  
antoniol@ieee.org

Licong Cui, Tingting Yu  
Computer Science Department  
University of Kentucky  
Lexington, Kentucky, USA  
lcui@cs.uky.edu,  
tyu@cs.uky.edu

Abstract— Context and motivation. Trace matrices are the lynchpin of verification and validation activities that must be performed for mission- and safety-critical software systems: criticality analysis, completeness analysis, change impact analysis, etc. Studies have shown that automated traceability techniques can achieve high recall and sometimes acceptable precision when used to generate trace matrices [1]. The human analyst is required in the loop for many critical software systems and plays a role in vetting the auto-generated trace matrices. Studies have shown that humans are fallible and tend to decrease the accuracy of auto-generated trace matrices [2, 3, 4]. To address the need for improved matrix quality and synergy with analysts, researchers are examining methods that have received popular and high acclaim. We surmise that “big data,” deep learning, and meta-heuristic search are three categories of interest. Big data refers to “an emerging data science paradigm of multi-dimensional information mining for scientific discovery and business analytics over large-scale infrastructure” [5]. In addition, when facing complex classification problems, deep learning [7] has proven to be effective [7,8]. However, not all data have been created equal, and some data are likely more important than others [11]; unfortunately, exhaustive search is oftentimes not feasible and we must resort to heuristic methods [10]. Problem statement. Automated trace link generation techniques suffer from low precision and lack of synergy with human analysts. There is a potential that big data technologies, deep learning, and heuristic optimization can assist with automated trace link generation due to enormous software artifacts data and its complex structures. Ideas and results. We plan to characterize trace generation in terms of an unbalanced big data classification problem. For example, we will examine the typical size of software engineering artifacts, software elements that comprise the artifacts, diversity of the datasets, granularity of the datasets, and align them with big data technique pre-requisites/requirements. Though it may appear that traceability datasets are not large enough to apply big data techniques, with software engineering artifacts generally consisting of thousands of elements versus millions or billions, we can borrow semantically reach words encoding from natural language processing techniques [8]. Possibilities for addressing this include increased granularity in order to expand the size of datasets, deriving more data elements featuring disparate aspects of the datasets, etc. However, new ideas are needed to properly handle the challenge of highly unbalanced datasets where only a handful of true links exist. We expect that expanding the size of the data could simplify the rebalancing problem and improve the accuracy of trace generation. A second possibility would be to formulate the classifier-rebalancing problem as a search problem [10] or to model the trace recovery as a classification task where deep learning techniques place true traces close in the feature space making the similarity between true links higher. Alternatively, we may use big data representations for trace elements such as directed acyclic graphs and perform concept mining over the graphs [6]. Contributions and future directions. Inspired by prior work [10,11,12], we plan to capitalize on big data approaches successfully applied to biomedical problems [6], heuristic optimization [10], and deep learning [7,8,12] and learn how to apply them to the trace link generation problem.

Keywords-big data; trace link generation; traceability

Acknowledgment

Work was supported by NSF grants CCF-1464032 and CCF-1511117.

References


