Experiment Workbench: A Co-agent for Assisting Data Scientists

Leonardo Guerreiro Azevedo, Raphael Melo Thiago, Marcelo Nery dos Santos
and Renato Cerqueira
IBM Research, IBM, Av. Pasteur, 146, Rio de Janeiro, Brazil

Keywords: Data Science, Big Data, Knowledge Intensive Processes.

Abstract: The analysis of large volumes of data is a field of study with ever increasing relevance. Data scientists is the moniker given for those in charge of extracting knowledge from Big Data. Big data is high-volume, high-velocity and high-variety information assets that demand cost-effective, innovative forms of information processing for enhanced insight and decision making. The exploration done by data scientists relies heavily on the practitioner experience. These activities are hard to plan and can change during execution – a type of process named Knowledge Intensive Processes (KiP). The knowledge about how a data scientist performs her tasks could be invaluable for her and for the enterprise she works. This work proposes Experiment Workbench (EW), a system that assists data scientists in performing their tasks by learning how a data scientist works *in-situ* and being a co-agent during task execution. It learns through capturing user actions and using process mining techniques to discover the process the user executes. Then, when the user or her colleagues work in the learned process, EW suggests actions and/or presents existing results according to what it learned towards speed up and improve user wok. This paper presents the foundation for EW development (*e.g.*, the main concepts, its components, how it works) and discuss the challenges EW is going to address.

1 INTRODUCTION

Data scientists work at extracting actionable knowledge from data sources that can be heterogeneous, unstructured, incomplete and/or very large. Increasing volume of data collected by companies make the role of data scientists vital to business success. Beyond the technical skills, data scientists must also be capable of using extracted knowledge as the driving force behind actual changes in business course (Davenport and Patil, 2012). The work of a data scientist can be categorized as *knowledge work* (Davenport, 2005), and the processes she performs as *Knowledge Intensive Processes* (KiPs).

KiPs are characterized by activities that cannot be easily planned, may change on the fly and are driven by the contextual scenario that the process is embedded in (Di Ciccio et al., 2012). This conjunction of factors presents an opportunity to improve data scientist's efficiency through providing tools that can transform tacit procedural knowledge into explicit knowledge that is useful, both to the individual data scientist and to enterprises at large.

This work proposes a Joint Cognitive System (JCS) called Experiment Workbench (EW). JCS is defined as the combination of human problem solver and the automation and/or technologies that must act as co-agents to achieve goals and objectives in a complex work domain (Hollnagel and Woods, 2005). The idea of a human-computer symbiosis is gaining momentum in the Cognitive Computing research (Kelly, 2015) where humans and computers collaborate, using their unique and powerful capabilities, to build an environment where knowledge is created and evolves over time considering environment events.

Experiment Workbench's goal is to learn how data scientists actually perform their daily activities to assist them in future cases. This might be done by capturing user actions in logs and using Process Mining techniques to discover the underlying process. Process mining uses data recorded in event logs to extract how the process was actually performed. Each event in the log refers to an activity (*i.e.*, a well-defined step in some process) and is related to a particular case (*i.e.*, a process instance) (Van der Aalst, 2013).

Given the nature of the work performed by data scientists, Experiment Workbench's goal is to support them in their tasks, being unable to replace the human factors associated with the KiPs. A close collaboration between data scientists and a JCS (our EW) is

Azevedo, L., Thiago, R., Santos, M. and Cerqueira, R.
Experiment Workbench: A Co-agent for Assisting Data Scientists.
DOI: 10.5220/0009489905880595
In *Proceedings of the 22nd International Conference on Enterprise Information Systems (ICEIS 2020) - Volume 1*, pages 588-595
ISBN: 978-989-758-423-7
Copyright © 2020 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved

likely to succeed than they working separately (Layton et al., 1994). Experiment Workbench will help to shine a light on how data scientists perform their work and also streamline their workflows by providing shortcuts for repetitive tasks: started manually by data scientists or automatically by Experiment Workbench itself.

The remainder of this paper is divided as follows. Section 2 presents a background of the main concepts: Data Scientist, Knowledge Intensive Process (KIP), and Process Mining. Section 3 presents Experiment Workbench requirements and a proposal for its architecture. Section 4 discusses the main challenges behind the use of Experiment Workbench in Data Scientist daily activities. Finally, Section 5 presents the conclusion, the current state of this research, and proposals of future work.

2 BACKGROUND

This section presents the background required to understand the proposal of this work.

2.1 Data Scientists and KDD

Data scientists' role is relatively new (circa 2008), therefore there is no consensus about the meaning of the term nor the responsibilities and boundaries of the role, besides the precise definition is not a pressing issue (Provost and Fawcett, 2013). However, it is valuable to understand how the role is related to other important concepts, like: "Big Data", "data analysts" and "data mining".

Big data is high-volume, high-velocity and high-variety information assets that demand cost-effective, innovative forms of information processing for enhanced insight and decision making[1]. The role of the data scientist arises as a necessity: how to bring forward knowledge from databases with ever increasing complexity - larger, more heterogeneous, less structured and incomplete data.

In this scenario, structured data analysis alone is not enough. There is a lot of knowledge to be extracted joining disparate data sources. Although tooling are almost the same (*e.g.*, data mining techniques), the insights should be more aligned to businesses and, more than that, they should present hypotheses and conclusions in a way compelling enough to change businesses courses. Hence, data scientists are responsible for capturing and propagating knowledge extracted from databases that may be big, unstructured, heterogeneous and may have incomplete data. Beyond their technical skills, data scientists have to understand how to tell a story compellingly enough to convince C level executives. Besides, the story should be strongly supported by actual data.

Our proposal, the Experiment Workbench, will assist data scientists in their data exploration activities. Experiment Workbench will need to understand which process the data scientist executes during exploration tasks. Figure 1 presents an overview of the "Knowledge Discovery and Data Mining" (KDD)[2] process (Fayyad et al., 1996). KDD process is interactive and iterative, and with multiple points where user decision influences execution. The KDD process has seven steps: (i) Understand the domain and gather contextual knowledge; (ii) Create a target dataset by selecting relevant data where discovery will be performed; (iii) Clean the data (*e.g.*, remove noise), gather other information, apply strategies to handle missing information; (iv) Find useful features to represent the data, reducing dimensionality and applying some transformation methods, effectively reducing the search space; (v) Choose the particular data mining methods, techniques and features that will be used to find patterns in the prepared data; (vi) Interpret mined patterns, *e.g.*, visualizing the pattern and the data underlying the extracted models; (vii) Incorporate the knowledge discovered into other systems or simply document the acquired knowledge for further actions.

CRISP-DM (CRoss Industry Standard Process for Data Mining) (Shearer, 2000) presents a slightly modified view of KDD's process (Figure 2), which starts with activities to understand the business and the data, which represents our goal to understand the activities a user perform to execute her work in data science. In Experiment Workbench context, our goal is to give different level of process abstractions, which may be framed like the four level breakdown of CRISP-DM, with variable granularity (Figure 3): from coarser grains (*i.e.*, more general), at the same level of KDD overview, to finer grains, getting closer to the actual tasks.

KDD process is defined in very broad terms. Each step in the process can be "implemented" in different ways. The process is unpredictable, *i.e.*, different instances of the process with the same goal can present very different execution paths. This fact classifies the KDD process as a Knowledge Intensive Process (KiP) (Eppler et al., 1999).

[1] http://www.gartner.com/it-glossary/big-data

[2] KDD acronym can also mean "Knowledge Discovery in Databases".

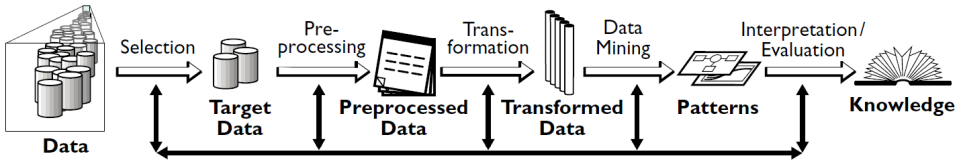

Figure 1: Overview of KDD process (Fayyad et al., 1996).

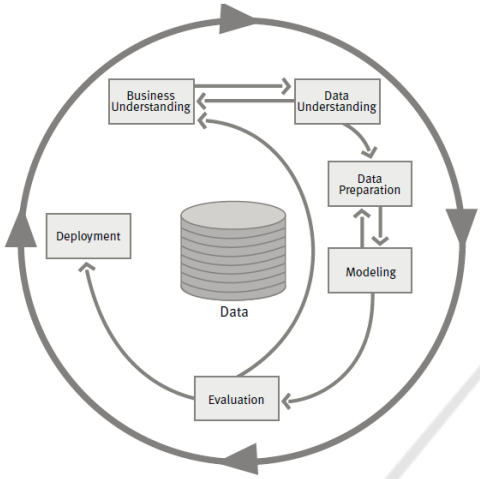

Figure 2: CRISP-DM (Shearer, 2000).

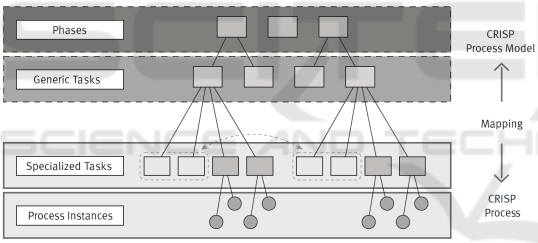

Figure 3: Four level breakdown of the CRISP-DM (Shearer, 2000).

2.2 Knowledge Intensive Processes

Business processes are core assets of organizations. They corresponds to "chains of events, activities and decisions". Business Process Management (BPM) has the goal to oversee how work is performed in an organization to ensure consistent outcomes and to take advantage of improvement opportunities (Dumas et al., 2013).

Knowledge Intensive Processes (KiPs) are a specific class of business processes characterized by "activities that cannot be planned easily, may change on the fly and are driven by the contextual scenario that the process is embedded in" (Di Ciccio et al., 2012). KiPs are knowledge- and data-centric, and require flexibility at design- and run-time (Di Ciccio et al., 2015). People that act in KiPs are called *knowledge workers*. They are workers that "think for a living" (Davenport, 2005). Data scientists are a specific type of knowledge worker.

Figure 4 shows the process spectrum. It relates knowledge intensity with predictability and structuring. As a general rule, structure, predictability and automation are inversely proportional to knowledge intensity. A process that is completely unpredictable and, therefore, non repeatable, is classified as an unstructured process, *i.e.*, no underlying rule governs how the process instance behaves. On the other hand, highly predictable processes are the structured ones, *i.e.*, new instances behave very orderly, following a set of predefined rules.

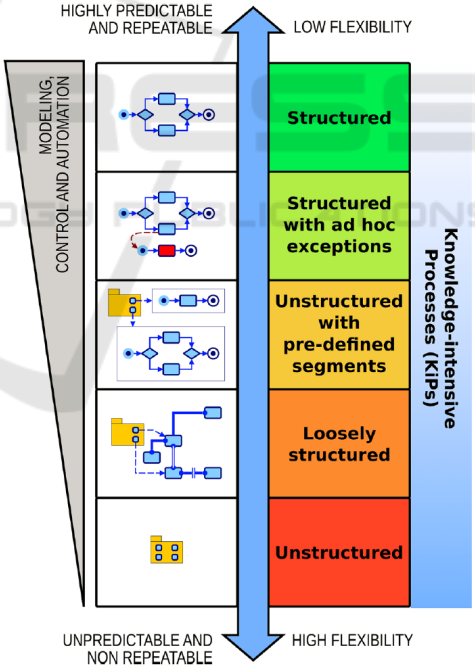

Figure 4: The spectrum of process management (Di Ciccio et al., 2012).

Using these categories, KDD's process overview seems to be a loosely structured process. One of Experiment Workbench's goal is to provide higher levels of structure to the data exploration process enacted by data scientists. We hypothesize that even within highly chaotic processes, experts use heuristics, de-

veloped in *ad hoc* manner, to achieve their goal (*intuition*). In our proposal, some of these heuristics could be represented using fragments of processes. Discovering such fragments would allow Experiment Workbench to pass KDD process from "loosely structured" to "unstructured with pre-defined segments", maybe even to "structured with *ad hoc* exceptions". This would depend on how mature the process actually is.

2.2.1 Process Mining

Process mining is a discipline that relates data mining and process intelligence. Process intelligence is the discovery, analysis and verification of process effectiveness in improvement of business (Dumas et al., 2013). Process mining is performed over data recorded in event logs, containing information that was created during the execution of the process (van der Aalst and Weijters, 2004). A process discovered through mining event logs is called *as-is* process, *i.e.*, it presents the process that really runs. On the other hand, a modeled processes is called *to-be* process, *i.e.*, it presents how the process should run (Dumas et al., 2013).

There are three approaches in process mining (Dumas et al., 2013): process discovery, process analysis and process verification. Mining the *as-is* process instances transforms tacit knowledge into explicit knowledge. These three approaches have the potential to help knowledge workers executing KiPs by partially automating the process and supporting the visualization of the *as-is* process. Even if full automation and instrumentation of the processes is not possible, it might be feasible to find fragments of processes which are stable, and automate them or make them explicit to be used as a source of learning by non-experts. Viewing the *as-is* process transforms a tacit knowledge into explicit knowledge, making the worker more aware of her own workflow.

3 EXPERIMENT WORKBENCH

The Experiment Workbench assists data scientists. Its main functionality is mine fragments of the exploratory process executed by data scientists. These processes are extracted from data captured by monitoring the interaction between data scientists and her tools. Experiment Workbench has three main distinct operation phases: (i) Monitor data scientist; (ii) Learn models; (iii) Apply models.

In Phase (i) (Monitor data scientist), Experiment Workbench collects data about the interaction of the data scientist with tools (*e.g.*, user-application events), and creates event logs which are stored in a provenance database. Provenance data helps track the derivation history of a data product (Simmhan et al., 2005), which allows answering questions like: "What activities were executed to achieve the result x?", "Who was responsible for producing the result y?", "Which were the parameters that generated data z?". Provenance data is an extra layer of information on top of event logs.

Information captured in Phase (i) is used in Phase (ii) (Learn models) to construct models (*e.g.*, processes or probabilistic models) that may help data scientists in future explorations. The main goal of this phase is to mine *as-is* processes, using provenance to create a predictive model that relates application features to process fragments (*i.e.*, sets of structured activities).

In Phase (iii), Experiment Workbench effectively helps data scientists by providing features, such as:

Feature 1. Visualization of *as-is* executions;

Feature 2. Query provenance data;

Feature 3. Composing existing functions;

Feature 4. Automating repeatable process fragments;

Feature 5. Preprocessing data and making it available on-demand, minimizing user waiting times;

Feature 6. Suggesting tasks to be executed.

Experiment Workbench is an assistant tool helping data scientists towards increased productivity and satisfaction. It does not have an objective to replace data scientist but rather working with her in a collaborative way. The ability to understand how the explorations are being executed (*Feature 1*) and querying provenance data (*Feature 2*) allows data scientists to have deep insight into their own work. Composing existing functions (*Feature 3*) allows the scientist to shortcut repetitive tasks, the same applies to the automation of some tasks (*Feature 4*). Experiment Workbench also processes data and makes it available to the user in a on-demand way (*Feature 5*). Lastly, Experiment Workbench may suggest relevant tasks to the data scientist (*Feature 6*), and she can choose the most appropriate course of actions.

Figure 5 presents an overview of Experiment Workbench's solution components. They are divided in three layers directly related with the three operation phases: Monitor data scientist; Learn models; and, Apply model.

In ***Monitor Data Scientist*** (P.1), information capturing is implemented by the following components: Functionality Register, Execution Listener and Interaction Tracker. *Functionality Register* is responsible

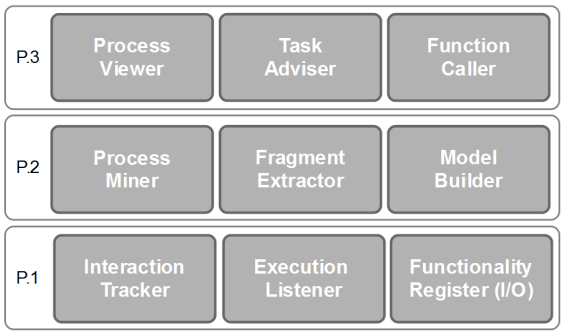

Figure 5: Experiment Workbench's architecture.

for managing (*e.g.*, gathering and storing) information about the systems used by the data scientist, *i.e.*, the systems' functionalities. Among this information is input and output data, and how the functionalities can be accessed, *e.g.*, the URL to access the service functionality. *Execution Listener* collects information regarding the interaction between the data scientist and the tools. Data gathered by *Execution Listener* are stored by *Interaction Tracker* in the provenance database. The *Execution Listener* uses mechanisms to intercept functionalities calls or the user include explicitly the call interceptor in tools' code using a library. Spring interceptor[3] is an example of mechanisms for the former case when the application is implemented as Java Servlet. For the latter case, the application developer uses a library that has functions to call the *Interaction Tracker* component to store the information needed to log the user events. Examples of tools that fit this case is ProvLake and Komadu in the multiworkflow, but they are focused in application code without necessarily gathering user actions with the system, but rather automatic workflow execution. ProvLake (Souza et al., 2019) is a tool that adopts design principles for providing efficient distributed data capture from workflows while Komadu (Gaignard et al., 2017)(Missier et al., 2010) is a distributed data capture solution that integrates provenance data in a multiworkflow execution. The *Interaction Tracker* receives log data, transforms it according to a provenance schema (*e.g.*, W3C Prov (Gil et al., 2013)) and stores it in the provenance database.

Model Learn (P.2) is performed by the components: Process Miner, Fragment Extractor and Model builder. *Process Miner* uses information gathered in P.1 to mine *as-is* processes, *i.e.*, to discover the process structure from the events. Some examples of tools that can be used to perform process mining are ProM (Van Dongen et al., 2005) and Apromore (La Rosa et al., 2011) (which are open-source), and Disco[4], Celonis[5] and ProcessGold[6] (which are commercial). *Fragment Extractor* uses discovered *as-is* processes to find highly correlated process fragments. The semantic of the process (*e.g.*, activities data, business rules, business requirements and roles) and the process' workflow patterns (van Der Aalst et al., 2003) (*e.g.*,, parallel split sequence, exclusive choice and simple merge) may be used to find the fragments. *Model Builder* extract rules to govern when a particular process fragment can be used.

Apply Model (P.3) is enacted through: Process Viewer, Task Adviser and Function Caller. The layer P.3 is the closest to the data scientist, being responsible to assist her. *Process Viewer* allows visualization of interactions as processes. *Task Adviser* advice on next steps to execute and, through an interaction with Process Viewer, presents the current execution as a process. *Task Adviser* can call *Function Caller* to automate execution of some fragments, it also presents results from these executions. *Function Caller* exposes found process fragments as functionalities, which are compositions of interactions with data scientist's tools.

4 DISCUSSION

This section discusses the main challenges the Experiment Workbench should address. The discussion is divided by challenges in each of the three operation phases.

4.1 Monitor Data Scientist

The availability of monitoring data related to the data scientist work is an important requirement for Experiment Workbench. This data represents the interaction between data scientists and the tools she uses. Monitoring presents different types of challenges: some psychological, others technical. Among the psychological challenges are:

P.I. The idea of constant monitoring may be resisted by users. The resistance may stem from a fear of having their workflow questioned;

P.II. Monitoring should concern not storing user sensitive data;

P.III. The user may change her behavior when she knows she is being monitored.

[3]https://docs.spring.io/spring/docs/current/spring-framework-reference/web.html#mvc-handlermapping-interceptor

[4]https://fluxicon.com/disco/

[5]https://www.celonis.com/

[6]https://processgold.com/

Technical challenges related to monitoring raises when analyzing data gathering options:

T.I. Modifying existing tools that may have closed source, which brings questions like:
   (a) If there is no access to the tools' source code, data capture is very difficult or even not possible. Interceptor mechanisms should be used, but still it should be possible to intercept the tool's functionalities execution;
   (b) If systems are managed by a third party, it requires negotiations and agreements between parties.

T.II. Monitoring the environment where the interaction between data scientist and the tools takes place, which requires:
   (a) Monitor all the different environments where interactions may happen which may require the use and instrumentation of sensors;
   (b) Understand environment semantics and capture environment state in proper ways besides tools functionalities, and be able to integrate environment and tools captured data.

T.III. The scientist inform the Experiment Workbench while performing her activities, which require:
   (a) More work to be done. Data scientist's cognitive load might already be high, or given the exploratory characteristics of the tasks being done, she may think it is not worth to formalize each step, while in fact, each step gives a little more insight regarding the data and the overall process.
   (b) This approach could be bothersome for the data scientist.

4.2 Learn Models

Assuming the monitoring phase captures all relevant data, the next phase is to learn what a data scientist does.

I. Using provenance data may increase the accuracy of process mining techniques. Provenance data may add another layer of information or semantics;

II. Some of the decisions made by data scientists are likely to be based on features appearing on the manipulated data. Generalizing this process of feature selection, aiming at automation, is a non-trivial task.

III. Visualization tools should be used to show to the data scientist what was learned, and give her the possibility to make adjustments, and give feedback on what was captured and what was learned. This feedback may trigger refactoring of the capturing and the learning techniques.

4.3 Apply Models

The application of the learned model has the following challenges:

I. Online classification of the current tasks might be too costly;

II. Call the functionalities of the tools used by the data scientist may not be easy due to closed code, hardware requirements, tools technology etc.;

III. Execute automatically a task on behalf of the user may rise privacy constraints;

IV. Completely autonomous execution of process fragments might consume a lot of resources. This can be prohibitive either or both computationally and economically.

5 CONCLUSION

This work presented the proposal of Experiment Workbench, a tool aiming at supporting data scientists, automating, suggesting and presenting a historical view of their daily tasks. The research behind Experiment Workbench is still in its initial state; however, this work presents the basis upon development.

We touched important definitions related to the problem: what is a data scientist; how her work is related to Knowledge Intensive Processes; ideas about using process modeling to represent both data mining activities as well as KiPs; and, how process mining could be used to solve the problem. We presented an overview of the main Experiment Workbench components and the main challenges it should address.

Experiment Workbench work is divided into three main phases: (i) Monitor data scientist work; (ii) Learn what she does; and, (iii) Apply the learned activities to help her in her daily tasks. To be able to support each phase many technologies and concepts should be used in the implementation of Experiment Workbench, such as:

(i) Provenance representation, capture, storage and query;

(ii) Process mining to learn the process the data scientist execute;

(iii) Cognitive Computing and Joint Cognitive systems to allow properly human-computer symbiosis;

(iv) Process automation and software/component composition to allow the combination of tools functionalities to automate user work;

(v) Human-Computer Interaction (HCI) which is a research area that investigates and explores human-computer symbiosis;

(vi) Visualization techniques and tools in order to provide the learned knowledge to the data scientist so she can properly understand what was learned and how it was learned, and be able to give feedback;

(vii) Recommendation systems or recommender systems which are software tools and techniques that provide suggestions to support users' decisions (Adomavicius and Tuzhilin, 2005).

HCI and AI expertise are been combined for some time (Grudin, 2009). The combination of HCI and AI research is been a concerned of major tech companies like Google[7], which launched an initiative to study and redesign the ways people interact with AI systems (Holbrook, 2017).

As future work, we are going to further evaluate solutions for the main challenges EW faces and implement the proposed architecture considering: the use of provenance data to improve process mining; extracting relevant process fragments from a pool of process instances; diminishing the usage obstacles - in particular those related with monitoring; connecting ideas of Cognitive System Engineering methods (Elm et al., 2008)(Bonaceto and Burns, 2006) to find points in the interaction that can be improved and how they could be improved.

We also aiming at performing case studies to evaluate the tool in O&G area. The evaluation is not an easy tasks due to the several challenges Experiment Workbench has to deal with. We believe study case fits this work because it is an empirical research in a real context and there are variables in the investigative pheomenon we do not know or we do not want to control. Evaluation in real scenario will allow observation of how data scientist engage in their tasks using Experiment Workbench. A case study approach suggests the use of several data sources to deep the investigation, which meet Experiment Workbench evaluation mainly because the disparate characteristics of its components (Pimentel, 2011)(Yin, 2015).

[7]https://ai.google/pair/

REFERENCES

Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, 17(6):734–749.

Bonaceto, C. and Burns, K. (2006). Using cognitive engineering to improve systems engineering. In *Manuscript submitted for presentation at the 2006 International Council on Systems Engineering Conference*.

Davenport, T. (2005). Thinking for a living.

Davenport, T. H. and Patil, D. (2012). Data scientist. *Harvard business review*, 90:70–76.

Di Ciccio, C., Marrella, A., and Russo, A. (2012). Knowledge-intensive processes: an overview of contemporary approaches. *Knowledge-intensive Business Processes*, page 33.

Di Ciccio, C., Marrella, A., and Russo, A. (2015). Knowledge-intensive processes: characteristics, requirements and analysis of contemporary approaches. *Journal on Data Semantics*, 4(1):29–57.

Dumas, M., Rosa, M. L., Mendling, J., and Reijers, H. A. (2013). *Fundamentals of Business Process Management*, volume 1. Springer.

Elm, W. C., Gualtieri, J. W., McKenna, B. P., Tittle, J. S., Peffer, J. E., Szymczak, S. S., and Grossman, J. B. (2008). Integrating cognitive systems engineering throughout the systems engineering process. *Journal of Cognitive Engineering and Decision Making*, 2(3):249–273.

Eppler, M. J., Seifried, P. M., and Röpnack, A. (1999). Improving knowledge intensive processes through an enterprise knowledge medium. In *Proceedings of the 1999 ACM SIGCPR Conference on Computer Personnel Research*, SIGCPR '99, pages 222–230, New York, NY, USA. ACM.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37.

Gaignard, A., Belhajjame, K., and Skaf-Molli, H. (2017). SHARP: Harmonizing and bridging cross-workflow provenance. In *European Semantic Web Conference*, pages 219–234. Springer.

Gil, Y., Miles, S., Belhajjame, K., Deus, H., Garijo, D., Klyne, G., Missier, P., Soiland-Reyes, S., and Zednik, S. (2013). Prov model primer. https://www.w3.org/TR/2013/NOTE-prov-primer-20130430/.

Grudin, J. (2009). Ai and hci: Two fields divided by a common focus. *Ai Magazine*, 30(4):48–48.

Holbrook, J. (2017). Human-centered machine learning. https://medium.com/google-design/human-centered-machine-learning-a770d10562cd. Accessed in February 14th, 2020.

Hollnagel, E. and Woods, D. D. (2005). *Joint cognitive systems: Foundations of cognitive systems engineering*. CRC Press.

Kelly, J. E. (2015). Computing, cognition and the future of knowing. *Whitepaper, IBM Research*, 2.

La Rosa, M., Reijers, H. A., Van Der Aalst, W. M., Dijkman, R. M., Mendling, J., Dumas, M., and García-Bañuelos, L. (2011). Apromore: An advanced process model repository. *Expert Systems with Applications*, 38(6):7029–7040.

Layton, C., Smith, P. J., and McCoy, C. E. (1994). Design of a cooperative problem-solving system for enroute flight planning: An empirical evaluation. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 36(1):94–119.

Missier, P., Ludäscher, B., Bowers, S., Dey, S., Sarkar, A., Shrestha, B., Altintas, I., Anand, M. K., and Goble, C. (2010). Linking multiple workflow provenance traces for interoperable collaborative science. In *The 5th Workshop on Workflows in Support of Large-Scale Science*, pages 1–8. IEEE.

Pimentel, M. (2011). Estudo de caso em sistemas colaborativos. *Sistemas Colaborativos (capítulo 25). Rio de Janeiro: SBC/Elsevier*.

Provost, F. and Fawcett, T. (2013). Data science and its relationship to big data and data-driven decision making. *Big Data*, 1(1):51–59.

Shearer, C. (2000). The crisp-dm model: the new blueprint for data mining. *Journal of data warehousing*, 5(4):13–22.

Simmhan, Y. L., Plale, B., and Gannon, D. (2005). A survey of data provenance in e-science. *ACM Sigmod Record*, 34(3):31–36.

Souza, R., Azevedo, L., Thiago, R., Soares, E., Nery, M., Netto, M., Brazil, E. V., Cerqueira, R., Valduriez, P., and Mattoso, M. (2019). Efficient runtime capture of multiworkflow data using provenance. In *Proceedings of 15th International Conference on eScience (eScience)*, pages 359–368.

Van der Aalst, W. M. (2013). Process mining in the large: a tutorial. In *European Business Intelligence Summer School*, pages 33–76. Springer.

van Der Aalst, W. M., Ter Hofstede, A. H., Kiepuszewski, B., and Barros, A. P. (2003). Workflow patterns. *Distributed and parallel databases*, 14(1):5–51.

van der Aalst, W. M. P. and Weijters, A. J. M. M. (2004). Process mining: A research agenda. *Comput. Ind.*, 53(3):231–244.

Van Dongen, B. F., de Medeiros, A. K. A., Verbeek, H., Weijters, A., and van Der Aalst, W. (2005). The prom framework: A new era in process mining tool support. In *International conference on application and theory of petri nets*, pages 444–454. Springer.

Yin, R. K. (2015). *Estudo de Caso: Planejamento e métodos*. Bookman editora.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS